1998

Pitch detection techniques for prototype waveform coding

Pasquale M.B. Gambino
University of Wollongong

Pitch Detection Techniques

for

Prototype Waveform Coding

A thesis submitted in partial fulfilment of the
requirements for the award of the degree

Master of Engineering

from

University of Wollongong

by

Pasquale M. B. Gambino (B.E.)

Department of Electrical, Computer & Telecommunications Engineering

1998

Abstract

Speech coding at the low bit-rate of 2400 bits per second requires an accurate and robust pitch period computed at low algorithmic delay to produce a perceptually satisfactory result. Low bit-rate speech coding algorithms such as those based on Prototype Waveforms can offer a speech quality comparable to higher bit rate counterparts. In this thesis two new algorithms, the 'Prototype Waveform Pitch Detector' (PWPD) and, the 'Dynamic Programming/Viterbi' (DP/V) Pitch Detector are introduced which can determine the pitch at frequent 5ms intervals. The PWPD is based on the 'Composite' Auto Correlation technique which sub-divides the speech frame into fixed sub-frames. The use of fixed sub-frames neglects to take full advantage of the variation in the pitch period. Within the paradigm of Prototype Waveforms, the PWPD achieves accurate pitch tracks by tracking the constituent autocorrelations across the speech frame, therefore maximising the detection of the pitch period and, at a significantly lower look-ahead delay. A further reduction in the look-ahead delay is achieved by the DP/V Pitch Detector which is based on the 'Principle of Optimality' with a substantial extension of a variable state Viterbi-type Trellis. The inclusion of the trellis provides the capacity to maintain multiple candidate pitch tracks from which an optimal track (based on accumulated scores) is chosen.

Table of Contents

ABSTRACT 2

TABLE OF CONTENTS 3

NOMENCLATURE 8

CHAPTER 1

INTRODUCTION 10

1.1 INTRODUCTION 11

1.2 THESIS ORGANISATION 13

1.3 CONTRIBUTIONS 16

1.4 ACKNOWLEDGEMENTS 17

1.5 PUBLICATIONS 17

CHAPTER 2

LITERATURE REVIEW 18

2.1 ANALYSIS OF THE SPEECH SIGNAL 19

2.1.2 Linear Prediction of the Speech Signal 20

2.1.3 Long-Term Speech Characteristic 23

2.1.4 Pitch Period Determination 25

2.2 REAL TIME PITCH DETECTOR 29

2.3 A PREVIOUS EVALUATION OF SEVEN PITCH DETECTORS 32

2.3.1 Comparative Performance 32

2.3.2 The Average Magnitude Difference Function 33

2.3.3 Evaluation Criteria 35

2.4 CORRELATION BASED PITCH DETECTION 36

2.4.1 Auto Correlation based Pitch Detection 36

2.4.2 Maximum Likelihood and Auto Correlation 37

2.4.3 Cross Correlation Pitch Determination 39

2.5 ATTAINING LOWER BIT-RATE PITCH PARAMETERS 40

2.6 SIMPLIFIED INVERSE FILTER TRACKING ALGORITHM 43

2.6.1 SIFT Introduction 43

2.6.2 The SIFT Algorithm 43

2.6.3 The SIFT Algorithm Implementation 45

2.6.3.1 Reduced Auto Correlation Function 45

2.6.3.2 Algorithmic Delay 45

2.6.3.3 Fundamental (F0) Presence 45

2.6.4 SIFT Algorithm Pitch Detector Results 48

2.7 GLOTTAL CLOSURE INTERVAL PITCH DETECTION 50

2.7.1 GCI Introduction 50

2.7.2 Determining the Instant of Glottal Closure 51

2.7.2.1 Maximum Likelihood Epoch Detection Signal 52

2.7.2.2 Glottal Closure Instant Selection Signal 53

2.7.3 Benefits of the Glottal Closure Interval Selection Signal 53

2.7.4 GCI Pitch Detector Implementation 54

2.7.5 GCI Pitch Detector Performance 55

2.7.6 GCI Pitch Detector Results 60

2.8 CHAPTER SUMMARY 62

CHAPTER 3

PROTOTYPE WAVEFORM PITCH DETECTION 64

3.1 PROTOTYPE WAVEFORM PITCH DETECTOR 65

3.1.1 PWPD Introduction 65

3.2 PROTOTYPE WAVEFORMS 69

3.2.1 Prototype Waveform Representation 69

3.2.2 Prototype Waveform Interpolation 71

3.2.3 Multi-Prototype Waveform Interpolation 72

3.3 SHORT-TERM COMPOSITE AUTO CORRELATION 73

3.4 PROTOTYPE WAVEFORM-BASED PITCH DETECTION 76

3.5 PROTOTYPE WAVEFORM PITCH DETECTOR IMPLEMENTATION 77

3.5.1 Integral Tracking 77

3.5.2 PWPD Composite Auto Correlation Function 81

3.6 PROTOTYPE WAVEFORM PITCH DETECTOR RESULTS 82

3.7 CONCLUSION 83

CHAPTER 4

DYNAMIC PROGRAMMING/VITERBI PITCH DETECTION 85

4.1 INTRODUCTION 86

4.2 DYNAMIC PROGRAMMING 88

4.2.1 Principle of Optimality 88

4.2.2 Optimal Solution Derivation 90

4.2.3 Reduced Form Dynamic Programming Algorithm 91

4.2.3.1 Avoiding Unwanted Signal Samples 93

4.2.4 A Slope Constraint 94

4.2.5 Dynamic Programming Distance Function 97

4.3 OPTIMAL PITCH TRACK DETERMINATION 98

4.4 HIDDEN MARKOV MODEL AND THE VITERBI ALGORITHM 102

4.4.1 Symbol Observation Space 102

4.4.2 Variable Number of States 103

4.4.3 Constrained State Transitions 104

4.4.4 Null State Transitions 106

4.5 STATE ACQUISITION AND WEIGHTING SCHEME 107

4.5.1 State Acquisition 107

4.5.2 State Probabilities 108

4.5.3 Weighting Functions 108

4.5.3.1 State Observation Weighting 109

4.5.3.2 State Transition Weighting 110

4.6 DYNAMIC PROGRAMMING / VITERBI MODEL 111

4.6.1 State Likelihood 111

4.6.2 Current States with a Source State 112

4.6.3 Current States without a Source State 113

4.6.4 Previous States to be Propagated 114

4.7 THE DP/VITERBI PITCH DETECTION ALGORITHM 115

4.8 OPEN LOOP OPERATION AND MEDIAN FILTER 119

4.9 DP/V PITCH DETECTOR RESULTS 120

4.10 CONCLUSIONS 121

CHAPTER 5

CONCLUSION **123**

5.1 COMPARATIVE RESULTS AND SUMMARY 124

5.2 CONCLUSION 126

5.3 CONTRIBUTIONS 130

5.4 FUTURE WORK 131

BIBLIOGRAPHY **132**

APPENDIX A - SIFT PITCH PROFILES **138**

APPENDIX B - GCI PITCH PROFILES **149**

APPENDIX C - PWPD PITCH PROFILES **160**

APPENDIX D - DP/V PITCH PROFILES **171**

Nomenclature

ACF	Auto Correlation Function
AMDF	Average Magnitude Difference Function
ARMA	Auto Regressive Moving Average
CCF	Cross Correlation Function
CELP	Code Excited Linear Prediction
DP	Dynamic Programming
GCI	Glottal Closure Interval
GSM	Global System Mobiles
HMM	Hidden Markov Model
ITU-T	International Telecommunications Union - Telecommunications
LP	Linear Prediction
LPAS	Linear Prediction-based Analysis-by-Synthesis.
LPC	Linear Prediction Coefficient
LPF	Low Pass Filter
LTP	Long-Term Prediction
MDPML	Multi Dimensional Pseudo Maximum Likelihood
MLED	Maximum Likelihood Epoch Detection
MPW	Multi Prototype Waveform
MSE	Mean Square Error
PEC	Partial Error Criteria
PML	Pseudo Maximum Likelihood
PW	Prototype Waveform

PWI	Prototype Waveform Interpolation
PWPD	Prototype Waveform Pitch Detection
Residual	Linear Prediction Residual
SIFT	Simplified Inverse Filter Tracking
Speech	Speech sampled at 8kHz
TFI	Time Frequency Interpolation
VA	Viterbi Algorithm

CHAPTER 1

INTRODUCTION

1.1 Introduction

Research into speech and audio digital signal processing techniques has dramatically advanced applications using digitally processed speech. The range of applications include Spoken Word Recognition, Speaker Verification Systems and Visual Aids (pitch profiles) for the aurally-impaired. This thesis presents the results of research into Low-Delay Pitch Detection, specifically targeted for low bandwidth 2400 bits per second (bps) speech coding algorithms for use across telecommunications networks. The significant reduction in transmission bandwidth can also benefit applications that use sophisticated encryption techniques. High quality 2400bps coding has been made possible by recent advancements in Low-Delay speech coding and these have highlighted the need for robust and accurate Pitch Detection. Two new algorithms are presented in this thesis, which can generate a robust and accurate pitch estimate at frequent 5ms intervals.

The analysis of the speech signal centres on its parameterization. To achieve this, the process involves the partitioning of the speech signal into two primary categories, the Vocal Tract parameters, and an Excitation sequence. The resonant frequencies generated by the Vocal and Nasal Tracts are termed the Formant Frequencies and are a characteristic of voiced speech. These frequencies vary with time in unison with the shape of the Vocal and Nasal Tracts. Once digitized, the speech signal is analysed for Short-Term correlation using Linear Prediction (LP) techniques, to obtain the Vocal Tract model. Voiced speech segments can be efficiently represented by a small set of LP coefficients over (typically) 20ms speech segments. This model representation of the Formant structure enables redundant data to be removed prior to transmission, which effectively reduces the bandwidth. The removal of redundant data, however, reduces the reconstructed speech signal quality, with distortions

such as false periodicity becoming more prevalent as the bit-rate is reduced. A basic speech generation model has a Noise component and a Pitch Period component (for voiced speech segments), as input signals. These two input signals are switched as required between Voiced and Unvoiced, prior to being Vocal Tract (Formant) filtered, for short speech segments as illustrated in Figure 1.

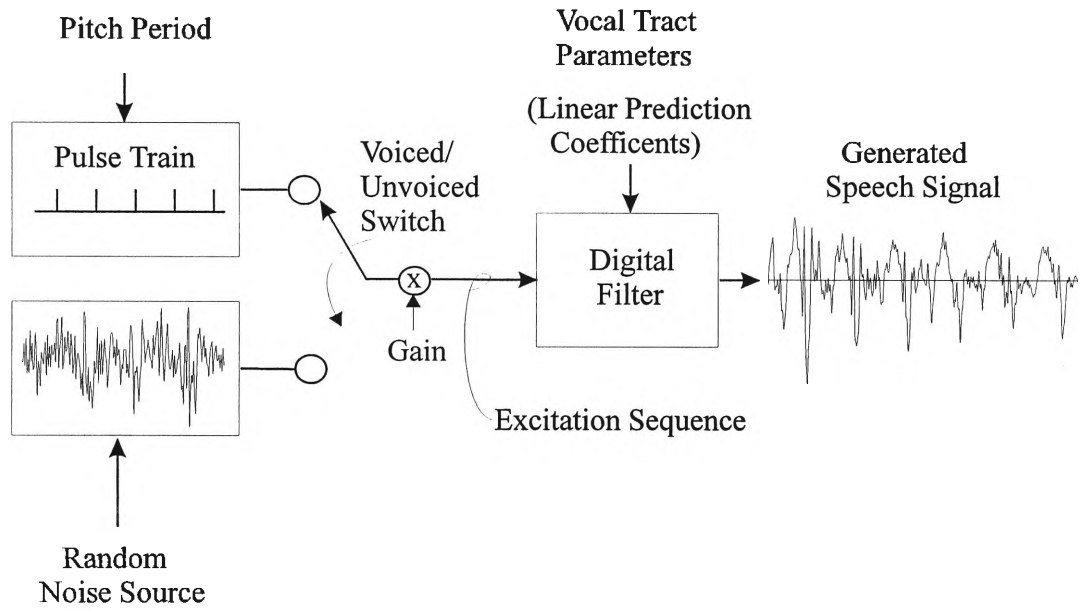

Figure 1 - Simplified Speech Generation Model [6-p398].

For Voiced speech segments the Pitch Period component models the Fundamental (lowest frequency) component by a suitable pulse train. This Long-Term pitch component is necessary, as it is not efficiently modelled by the Formant Filter. For speech segments which are Unvoiced, only the Noise component is Formant filtered, as these segments do not contain the resonant frequencies associated with voiced speech. The Excitation Sequence (including the Pitch Period component) models the error resulting from the Linear Prediction (termed the residual).

The detection of the pitch period within a short time period is an important attribute required by most low bit-rate speech coders. The accurate determination of the pitch period is considered vital if the quality of the reconstructed speech is to match, or supersede, its higher bit-rate counterpart. The greater the accuracy of the pitch period, the closer the resulting decoded speech will be to the original speech. Recent developments in low bit-rate speech coding have rekindled some of the difficulties associated with accurate and robust pitch period detection. Applications such as Speaker Verification and Word/Digit recognition systems are less stringent on evaluation criteria such as the algorithmic delay, which is important to low bit-rate speech coding. For instance, the delay in detecting the pitch period, comprised of inherent look-ahead and/or algorithm processing delay, is perhaps the most important criteria, besides accuracy with respect to a reference pitch track. The objective for a Low-Delay speech coder is to achieve an overall targeted total one-way delay of 90 ms [1],[2]. As a result of the Pitch Detection research undertaken in this thesis a selection of current time-domain pitch detection algorithms are revisited in Chapter 2, and two alternative techniques specifically developed for Low-Delay speech coding are presented in Chapter 3 and 4. The motivation for this study originated from the requirement to achieve robustness, accuracy, and low algorithmic delay characteristics.

1.2 Thesis Organisation

This thesis commences with a Literature Review, presented in Chapter 2. Initially, an early Real-time Pitch Detector, based on the Auto Correlation Function is reviewed. This algorithm implements an effective pitch detector by non-linearly pre-processing the speech signal to remove unwanted signal components that degrade the detection of the Fundamental

frequency. These unwanted signal components include the Formant frequencies, and the harmonics of the Fundamental component. Following this, two algorithms are implemented that were initially considered as candidates for Low-Delay pitch detection. The Simplified Inverse Filter Tracking (SIFT) Algorithm and the Glottal Closure Interval (GCI) Pitch Detector are presented, providing a baseline for the two new algorithms presented in Chapter 3 and 4. One of the benefits of the SIFT and GCI techniques is that they can accurately detect pitch epochs by enhancing the signal periodicities. Their respective algorithmic delays (an entire frame), however, do not meet the requirements for Low-Delay speech coders.

Prototype Waveforms (PW) and Waveform Interpolation (WI) techniques are considered as candidates for low bit-rate (2400b/s) speech coders [1],[2],[3]. Speech which is reconstructed using such techniques provides for a significant bit-rate reduction, while achieving the quality of higher bit-rate counterparts. The basis of the PW technique is that prototypes (pitch period in length) are extracted from the speech (or residual) signal and then parameterized using the Discrete Fourier Transform. This parameterization, therefore, permits linear interpolation of the prototypes, significantly reducing the required bandwidth. Prototypes can be extracted from the raw speech signal, however, the Linear Prediction residual signal is preferred. The Prototype Waveform Pitch Detection (PWPD) and the Dynamic Programming/Viterbi (DP/V) Algorithms, in Chapters 3 and 4 respectively, are considered suitable for PW coders. They provide two intra-frame based pitch detection methods whereby the pitch period is extracted at frequent (5ms) intervals.

The first new algorithm, the PWPD method, which is based on existing Short-Time Auto Correlation Function techniques [4], achieves accurate pitch tracks by considering pitch period detection in the paradigm of Waveform Interpolation (WI) [4],[5]. This is achieved by

tracking the constituent autocorrelations within a speech frame, thereby maximising the detection of the pitch period variation. The algorithm is capable of maintaining smooth pitch tracks at a significantly lower look-ahead delay than that required by alternative autocorrelation techniques.

The second new algorithm, 'Dynamic Programming/Viterbi (DP/V) Pitch Detection', is presented in Chapter 4. This algorithm is based on Dynamic Programming (DP) with a substantial extension incorporating a Viterbi-type trellis. DP has been used successfully for applications such as Spoken Word Recognition and Speaker Verification Systems and, to a limited extent, Pitch Detection. The strengths of DP lie in its signal-matching capabilities. This is achieved by applying the 'Principle of Optimality' and an associated error criteria to determine the optimal match between two similar speech segments. A deficiency arises, however, concerning the selection of non-optimal tracks. DP implementations can introduce a significant look-ahead delay to allow for corrections to the final track. This look-ahead delay would normally eliminate the DP Pitch Detectors from consideration for use in Low-Delay speech coding algorithms. The addition of a Viterbi-type trellis in the DP/V algorithm, however, has provided a robust open-loop scheme reducing the ambiguity in the pitch period detection while achieving smooth pitch tracks at a reduced look-ahead delay. The inclusion of a Viterbi-type trellis retains a number of likelihood pitch tracks from which an optimal track is then chosen. In achieving this, the Viterbi trellis survivor path determination process has been extended by the use of a Hidden Markov Model (HMM). The HMM, as a base, will account for pitch track discontinuities in the speech signal by providing mechanisms whereby candidate pitch tracks can be propagated within the trellis. The results of the DP/V algorithm presented in this thesis have demonstrated a reduction in the occurrence of pitch doubling or tripling.

1.3 Contributions

In summary the contributions claimed in this thesis are:

1.) A review of Pitch Detection methods, in Chapter 2, have highlighted the difficulties in determining accurate pitch estimates. The techniques vary from non-linear clipping operations performed on the speech signal to reduced-order linear prediction in order to successfully remove unwanted signal components that degrade pitch period detection. Implementations of the Simplified Inverse Filter Tracking algorithm and the Glottal Interval Closure Detection algorithm, are also presented in Chapter 2. These methods have demonstrated that enhancements to the speech signal periodicity contribute significantly to pitch period detection.

2.) The Short-Term Auto Correlation Function is the basis of the Prototype Waveform Pitch Detector (PWPD) presented in Chapter 3. Within the paradigm of Prototype Waveforms, a 'Composite' Auto Correlation Function, with an integral tracking capability, is introduced. This technique is better suited to tracking the pitch variation within a frame and, consequently, a reduction in the look-ahead delay is achieved providing a robust algorithm for the extraction of 'prototypes'.

3.) The algorithmic delay incurred by the variety of Pitch Detection algorithms, reviewed and presented in this thesis, is important to the success of a Low-Delay speech coder. The Dynamic Programming/Viterbi (DP/V) Pitch Detector, presented in Chapter 4, provides for a Low-Delay Pitch Detector capable of extracting an accurate pitch estimate at frequent (5ms) intervals while incurring a 12.5ms delay. This is achieved by retaining a number of Dynamic Programming time-warped plane minima expressed as paths within a Viterbi-type trellis.

1.4 Acknowledgements

I wish to thank, foremost, Dr Ian S. Burnett and the University of Wollongong for providing me with the opportunity and motivation to undertake the research into Speech Pitch Detection presented in this Thesis. The discussions with Ian provided me with great enthusiasm, without which this undertaking would not have been possible. I would also like to acknowledge the Commonwealth of Australia Department of Defence - HMAS Waterhen and, the Australian Postal Corporation - New South Wales, who have provided support throughout the duration of my studies.

1.5 Publications

1. I. S. Burnett, P. M. B. Gambino, *Pitch Detection based on Prototype Waveforms*; Proceedings of the Fourth International Symposium on Signal Processing and its Applications, Gold Coast Australia, 26th-30th Aug., 1996.

2. P. M. B. Gambino, I. S. Burnett, *Low Delay Pitch Detection using Dynamic-Programming/Viterbi Techniques*; Proceedings of the Fourth International Symposium on Signal Processing and its Applications, Gold Coast Australia, 26th-30th Aug., 1996.

3. P. M .B Gambino, I. S. Burnett, *A Low-Delay Dynamic-Programming/Viterbi Pitch Detector*; to be submitted for Publication.

CHAPTER 2

LITERATURE REVIEW

2.1 Analysis of the Speech Signal

'The object of speech analysis is to estimate the parameters of a speech model and their variations with time' [6]. The frequency spectrum of human speech is a characteristic of an individual's physiological formation of internal air-passages (vocal tract). This frequency spectrum is also determined by associated organs, such as the mouth, teeth and lips [7]. The vocal tract, and other chambers within the human respiratory system, form a complex sound wave guide (the larynx) through which air expired from the lungs travels, producing the desired sounds. Included amongst these are temporary chambers, created primarily by the use of the velum and aided by the tongue. The velum, located at the upper, rear of the mouth, is lowered, acting as a control valve directing the air flow to the nasal tract in lieu of the oral tract (mouth, teeth and lips). This is readily evident when pronouncing such nasal voiced speech that contain the consonant "n", as in "no". The dynamics of both oral and nasal tracts are the primary causes that generate the time-variant and non-causal characteristics of the speech signal. An example of the variation with time of a voiced speech signal is shown in Figure 2.1.

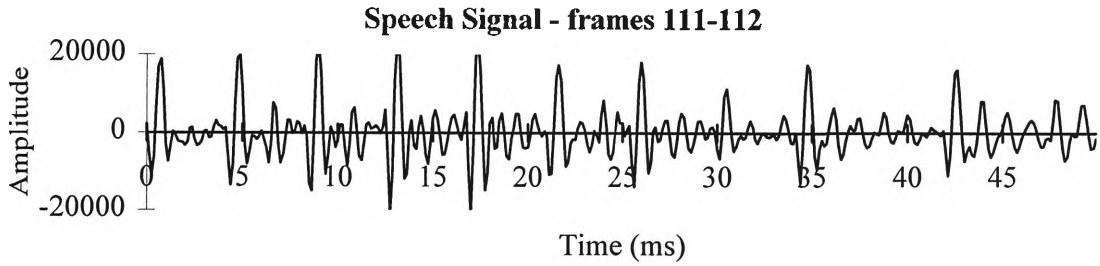

Figure 2.1 - A Voiced speech segment, representing 'a' in 'cats', spoken by a female, illustrates the time variation in Fundamental period.

Linear Prediction (LP) techniques are applied to relatively short speech segments in modelling the speech signal [6]. Within short analysis segments, typically 20-25ms, the speech waveform is assumed to be sufficiently stationary that LP techniques can be used with great success. A speech signal sampled at 8-10kHz is subsequently divided into 160-200 sample frames and prediction of the speech signal is performed over these reduced 20-25 ms intervals.

2.1.2 Linear Prediction of the Speech Signal

Assuming that speech signal is linear and stationary over short segments, it is possible to predict the speech signal. Within this interval the time domain representation is therefore given by:

$$\tilde{s}(n) = \sum_{k=1}^{P} a(k)\, s(n-k) \quad \text{..... (2.1)}$$

where $\tilde{s}(n)$ is the estimate of the speech signal, $s(n-k)$ is the input speech signal and $a(k)$ are the linear prediction coefficients.

The *'n'th* sample is estimated from *'P' (typically 10)* previous speech samples [6]. Once the predicted speech, $\tilde{s}(n)$, is computed it is subtracted from the corresponding raw speech signal, yielding an error signal, $e(n)$, or linear predicted residual $r(n)$ using:

$$e(n) = r(n) = s(n) - \tilde{s}(n) \quad \text{..... (2.2)}$$

and substituting, $\tilde{s}(n)$, with (2.1) obtaining the following expression for the residual,

$$r(n) = s(n) - \sum_{k=1}^{P} a(k)\, s(n-k) \quad \text{..... (2.3)}$$

A sample of the resulting residual waveform across two speech frames spanning 50 ms is shown in Figure 2.2.

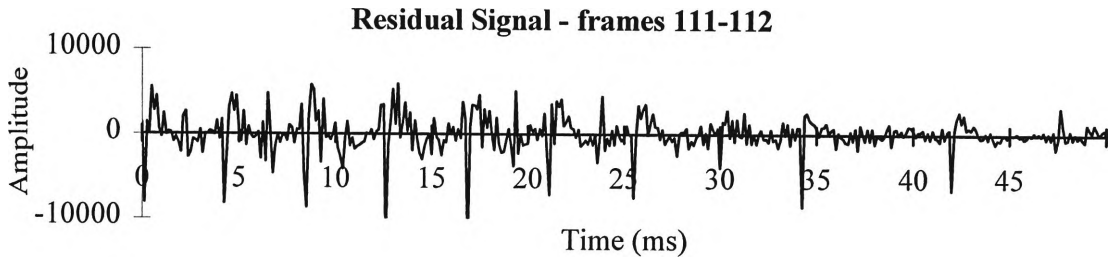

Figure 2.2 - Linear Prediction residual signal illustrating the distinctive glottal epochs, and the variation in adjacent peak positions with time.

When transformed into the Z-transform domain, equation (2.3) becomes:

$$E(z) = S(z).\, A(z) \quad \text{..... (2.4)}$$

where,

$$A(z) = 1 - \sum_{k=1}^{P} a(k)\, z^{-k} \quad \text{..... (2.5)}$$

In modelling the short-term speech signal by linear techniques, an all-pole system is used [8]. This also implies a minimum-phase system. In practice, however, nasal couplings introduce anti-resonances (zeros in the transfer function) [6] which are not modelled by such a system (this would require an ARMA model).

To determine the Linear Prediction (LP) coefficients, *a(k)*, minimisation of the mean-square error signal, *e(n)*, results in a set of linear equations which are termed the 'Normal Equations' [6]. In solving these Normal Equations, several efficient autocorrelation methods exist. Examples of these are the Levinson-Durbin and Schur algorithms which exploit the Toeplitz property of the autocorrelation matrix [6]. Once the LP coefficients are determined, the speech is filtered by a Finite Impulse Response (FIR) filter on a frame-by-frame basis. This filtering process is also referred to as 'Inverse Filtering' as it removes the short-term (Formant) structure of the speech waveform, whitening the speech spectrum. The major components in Linear Prediction Analysis are shown in Figure 2.3.

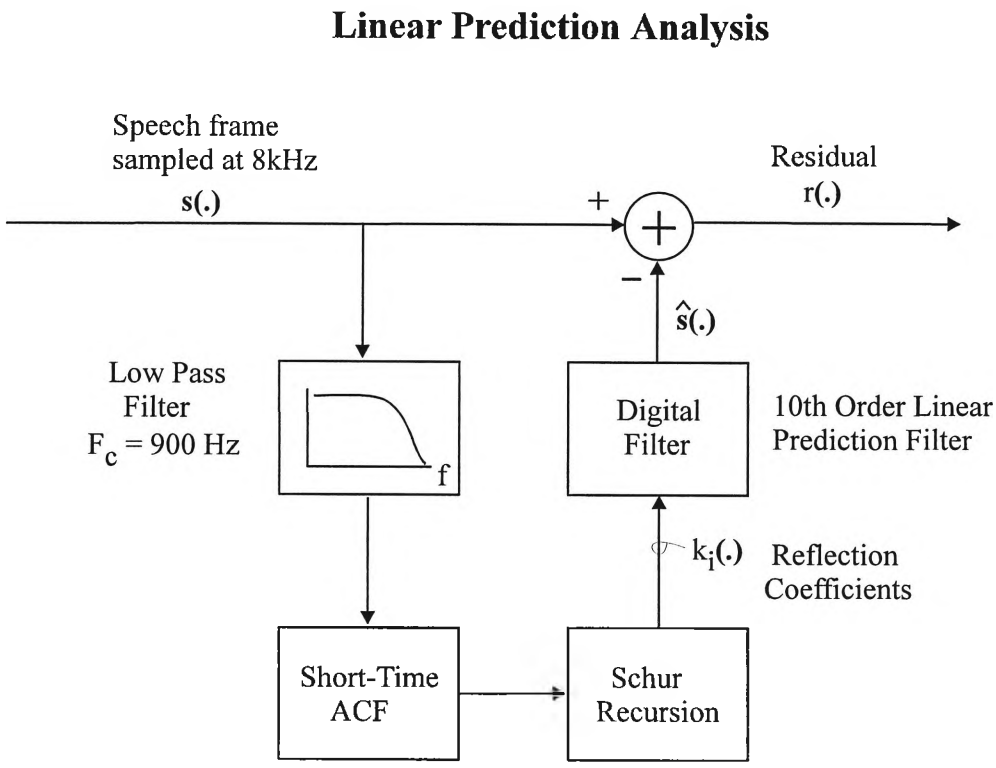

Figure 2.3 - Simplified Linear Prediction Analysis block diagram.

2.1.3 Long-Term Speech Characteristic

A vital second parameter that is not catered for in the Short-Term modelling is a Long-Term (LT) speech characteristic. This is illustrated by the Linear Prediction (LP) residual signal in Figure 2.2, where a distinctive pulse-like waveform is observed. This pulse train is a result of the inability of the Short-Term LP to model the LT component [6]. This Long-Term characteristic changes at a slower rate than the Short-Term Formant structure, accounting for the limitation of LP filtering. A significantly higher-order model would be required to predict such a pulse train (as for example, in G.729 LD-CELP a 50th order LP filter is used without LT filtering) [9]. In the context of the speech signal, the pulse-like waveform exhibited by the residual signal is representative of the Pitch or Fundamental (F0) frequency. Alternatively, the Pitch is the lowest recorded frequency of the speech waveform.

The glottis (gap between the vocal chords) is considered to be the source of the pitch pulses [7], and allows the expulsion of air from the lungs to separate the vocal cords, causing them to vibrate. Sudden opening of the glottis will occur provided an air pressure differential is present. All subsequent vocal cord vibrations, including vocal tract reflections, are modulated by the remainder of the vocal and nasal tracts, producing the final sound output (resonances). The pitch period is measured as "the elapsed time between two successive glottal pulses", or the closed glottis interval ('the pitch interval'), which may vary (+/-) 10% between successive periods and rarely exceeds 25% [11].

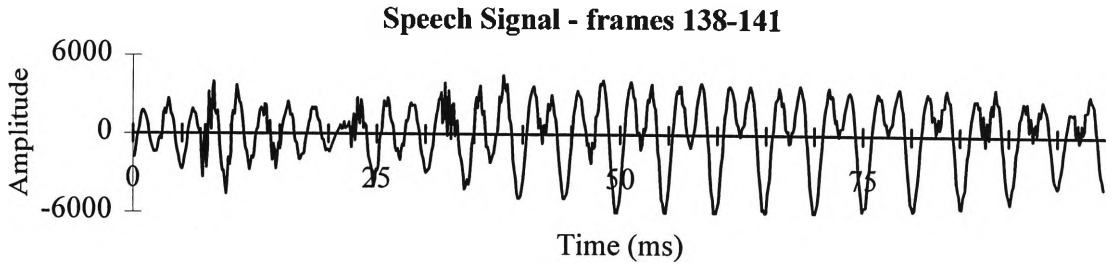

Figure 2.4a - Speech segment from 'e' in 'each' spoken by a female speaker. The time-sequence clearly illustrates the Fundamental component.

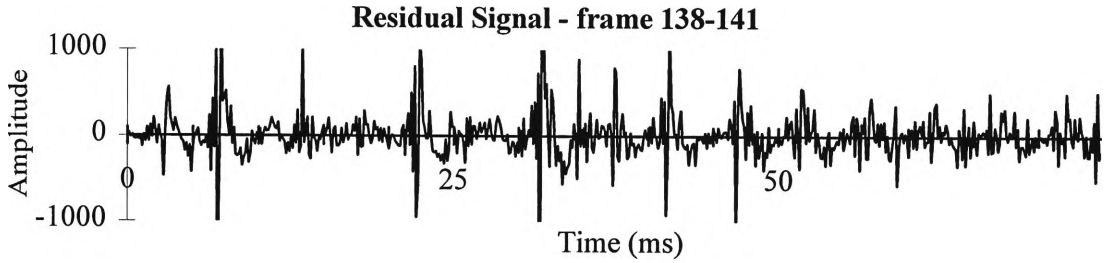

Figure 2.4b - A time-sequence illustrating both the variation between high energy residual spikes and the absence of distinctive pulses.

Figures 2.4a-b illustrate speech, and the corresponding residual, produced by a high-pitched (female) speaker pronouncing 'e' in the word 'each', spanning four contiguous 200 sample frames. This segment clearly illustrates the ambiguity in pitch pulse (epoch) positions. Additionally, towards the end of the residual segment there is a region in which the distinctive pitch pulses are non-existent. Pitch detection methods that rely on the residual signal are prone to fail in such circumstances [7].

2.1.4 Pitch Period Determination

In the majority of pitch detection algorithms the speech or residual signal is Low-Pass filtered (cut-off frequency 0.9-1kHz), removing higher Formant frequencies. The filtered signal will be comprised, predominantly, of the Fundamental (F0), accounting for the male and female F0 falling within the 50-400Hz frequency range. The remaining signal, however, will not exclude higher Formant components that fall within this bandwidth. A significant enhancement of the F0 is achieved by LP filtering and, as a consequence, the noise bandwidth is significantly reduced.

Pitch detection techniques that are based on the residual have resulted in improved performance due to the significant reduction in Formant components [7]. An ambiguity still persists however, with the occurrence of pitch period doubling or tripling, in which a harmonic of the Fundamental or a higher Formant frequency is dominant. To overcome this phenomenon pitch detectors introduce energy thresholds which are applied to secondary candidates if they are to supersede initial estimates. In addition, certain algorithms delay the final decision, permitting corrections to previous estimates in order to achieve smooth pitch tracks. A distinctive example of this ambiguity is that of a Diplophonic speaker, whereby the pitch may vary drastically between alternate pitch pulses and adjacent pitch pulses [7]. Secondary pitch period candidates are illustrated in Figures 2.5a-j, where the influence of the Formant components is observed. The ambiguity is further compounded by the rapid rise in pitch period. As an example, the transition between vowels 'a' and 'e' results in a rapid increase in the pitch period as illustrated in Figure 2.5f. Hence, the algorithmic delay incurred by pitch correction procedures is to the detriment of achieving high quality Low-Delay speech coders.

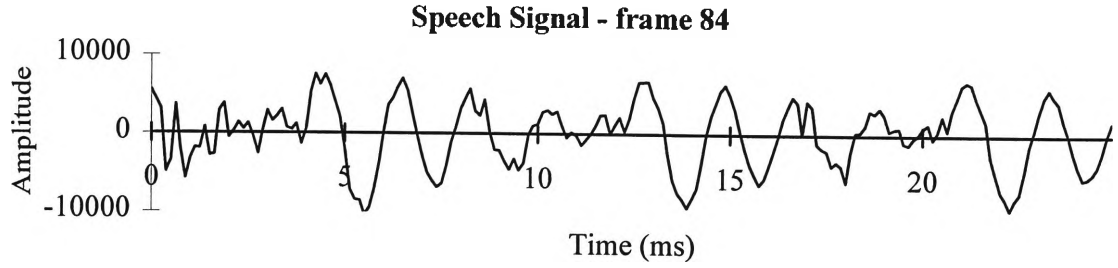

Figure 2.5a - First speech segment 'a' from 'shade' (male speaker).

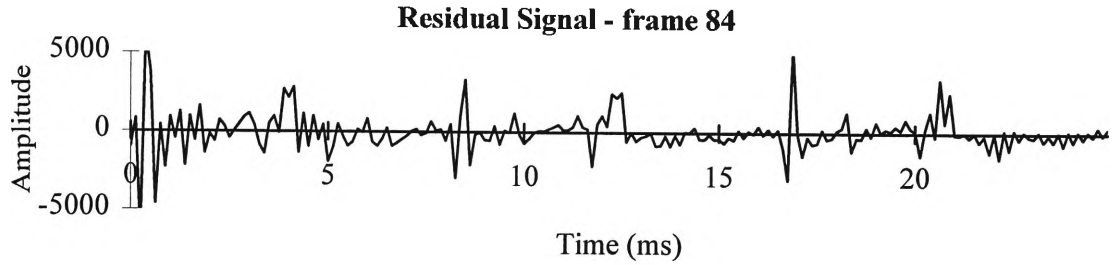

Figure 2.5b - First residual segment 'a' from 'shade' (male speaker).

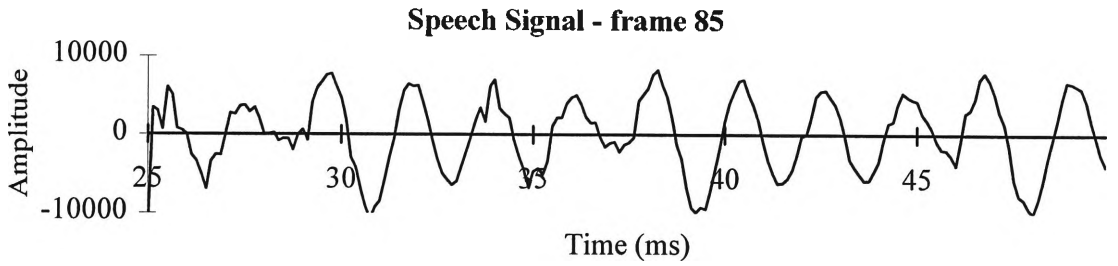

Figure 2.5c - Second speech segment 'a' from 'shade' (male speaker).

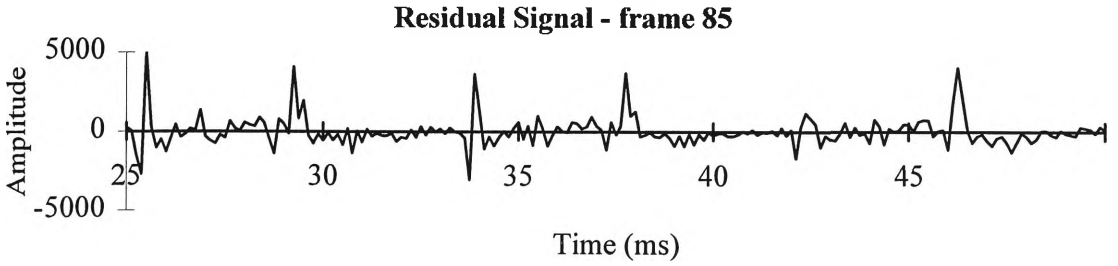

Figure 2.5d - Second residual segment 'a' from 'shade' (male speaker).

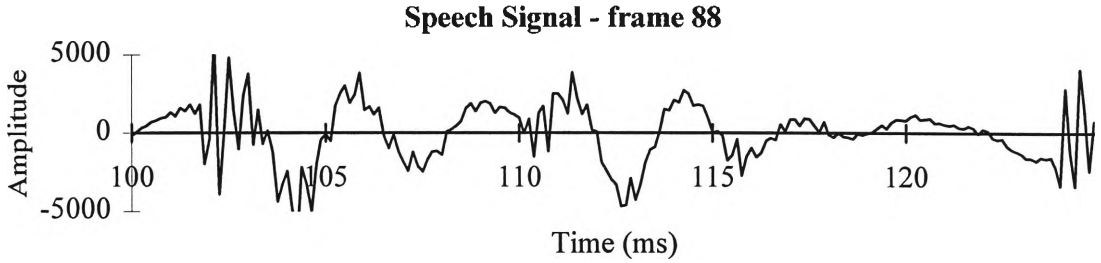

Figure 2.5e - First speech segment 'e' from 'shade' (male speaker).

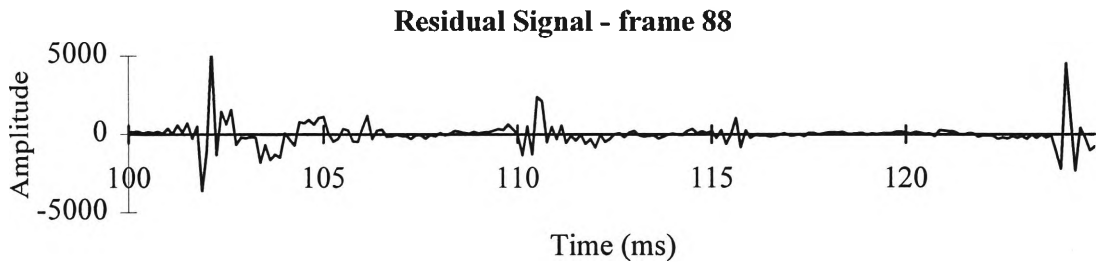

Figure 2.5f - First residual segment 'e' from 'shade' (male speaker).

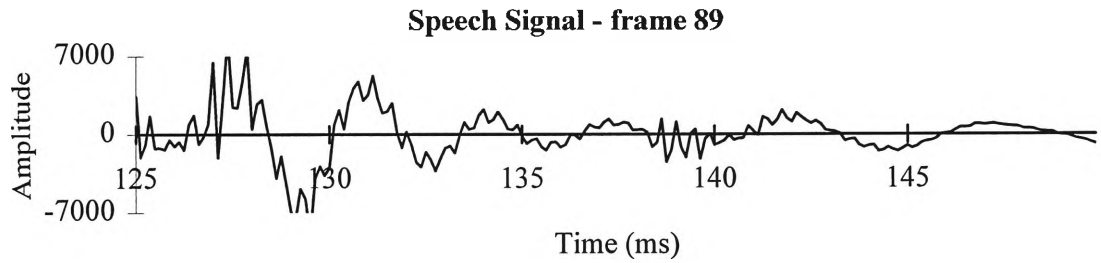

Figure 2.5g - Second speech segment 'e' from 'shade' (male speaker).

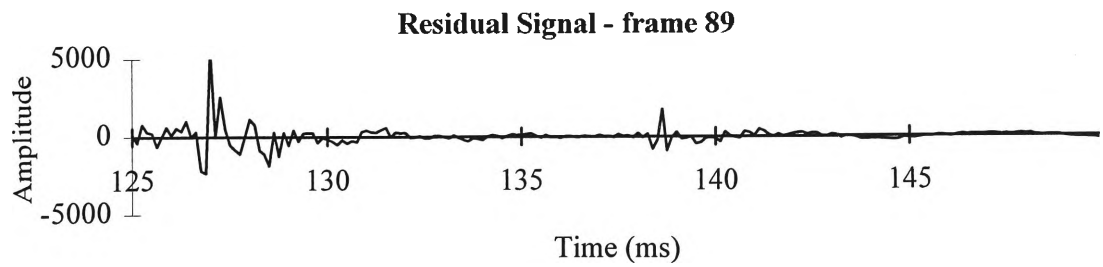

Figure 2.5h - Second residual segment 'e' from 'shade' (male speaker).

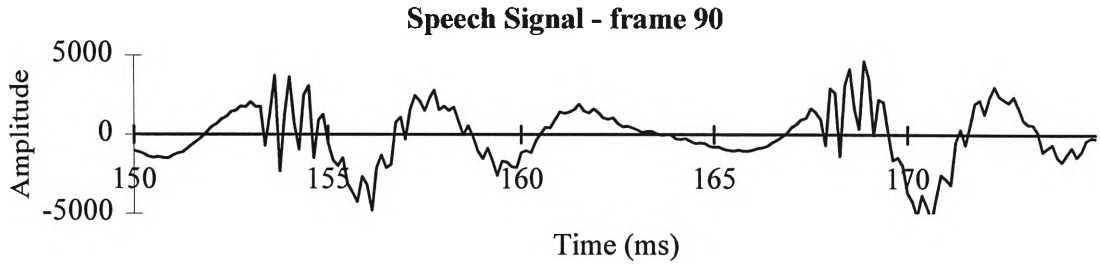

Figure 2.5i - Third speech segment 'e' from shade (male speaker).

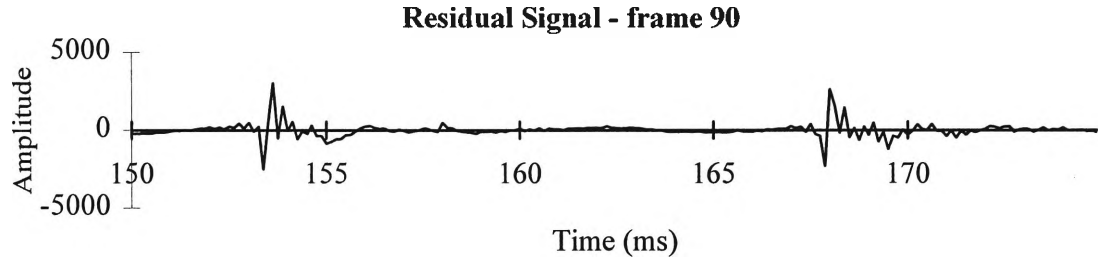

Figure 2.5j - Third residual segment 'e' from shade (male speaker).

2.2 Real Time Pitch Detector

The Real Time Pitch Detector presented by Dubnowski *et al.,* [13] in 1976, is based on the Clipped Auto Correlation Function (ACF). This technique is revisited as an introduction to Pitch Detection. The method is based on one of three techniques presented by Sondhi [14]. The original method proposed a non-linear, centre-clipping operation to be performed on the raw speech waveform. This operation is aimed at reducing the influence of Formant frequency components and their harmonics to the detection of the Fundamental frequency. Ideally, the signal that remains will pre-dominantly contain the Fundamental component, which is then extracted by ACF techniques.

The consequent removal of secondary pitch period candidates significantly enhances the detection of the Fundamental component in the subsequent Auto Correlation Function (ACF). In [13], the appropriate clipping threshold was found to be 80% of the smaller of two maximum peak signals recorded for the first and third 10ms sections of the 30ms frame. The speech signal was both centre-clipped and infinite-peak clipped, which resulted in a counter-type (+/-1) ACF method allowing real-time capability. The choice of both centre-clipping and infinite-clipping relate to the indiscriminate behaviour of the clipping functions. In applying centre- clipping solely, it is difficult to ascertain the validity of signal samples that either just exceed the threshold, or exceed it by a large margin [13]. An advantage of this technique is that when speech is High-Pass filtered (cut-off 200-300 Hz), due to the High-pass filter effect of the transmission channel, the Fundamental may not be present. An example of this would be a high pitched female speaker across a telephone channel. Sondhi states that "the absence of a large number of harmonics clearly is not a serious problem for the centre-clipping

method" [14] , however this absence of harmonics would be problematic for frequency based techniques.

Sample speech frames in Figure 2.6 (+/- thresholds indicated) illustrate the clipping function. Speech samples exceeding the threshold are retained, otherwise the signal sample is zeroed.

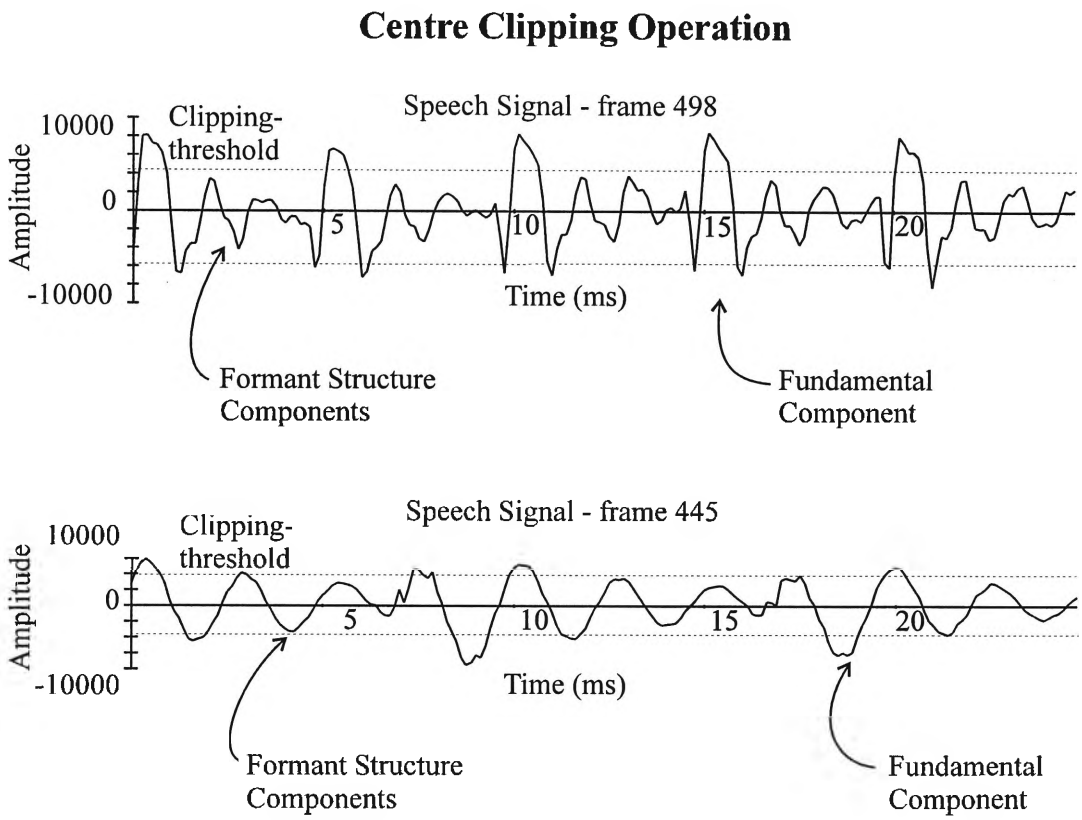

Figure 2.6 - Diagram illustrating the clipping threshold applied to two (typical) voiced speech frames and the removal of Formant frequency components.

The two frequency methods proposed by Sondhi, which reduced the influence of the formant structure, used the 'Spectrum Flattening' technique [14]. Spectrum flattening was achieved by a bank of bandpass filters with full-wave rectification, to produce a Short-time estimate of the signal frequency magnitude. In [14], a minimum phase correction procedure was used to

synchronise the harmonics at the various filter delays to maintain a linear phase response. The methods proposed by Sondhi all performed an Auto Correlation Function (ACF) on the spectrum-flattened output, and subjected the ACF to a peak-picking process in order to obtain the final pitch period. In computing the ACF, the Real-Time Pitch Detector [13] utilised overlapping 30ms (250-3250Hz) speech frames, where the overlap is 20ms to avoid discontinuities. The ACF is calculated to account for lags up to 20ms and normalised by the energy in the signal (ACF evaluated at zero delay).

Voiced-Unvoiced decisions were an integral component in the Dubnowski Pitch Detector [13]. If the ACF maximum exceeded a threshold of 30% relative to the frame energy, then the segment was classified as voiced, and the pitch period was recorded as the position of this maximum. Silence classification was determined by recording the peak signal level over 50ms of background noise in order to determine an appropriate threshold. If the peak signal level within a 30ms frame did not exceed this threshold, then the frame was classified as silence.

Reported disadvantages [6] with the non-linear clipping operations relate to the long analysis frame, which incurs a relatively long look-ahead delay. Additionally, the final pitch decision is subjected to error-correcting procedures that require preceding pitch estimates to be taken into consideration. These error-correcting procedures will delay the final pitch decision, adding to the overall delay.

2.3 A Previous Evaluation of Seven Pitch Detectors

A comprehensive evaluation of seven Pitch Detectors in [15] revealed the difficulties associated with pitch determination, and highlighted certain factors pertaining to the speech signal structure that need to be taken into consideration when evaluating a prospective algorithm. These factors included: (1) the time-varying glottal pulse train, both in structure and occurrence of the glottal epochs; (2) the Formant structure of the speech waveform, which is superposed on the Fundamental component, and is detrimental to reliable pitch detection; (3) the removal of low frequency components (0-300Hz) caused by the telephone (transmission) channel which acts as a high-pass filter, and subsequently the Fundamental which falls within this bandwidth is, therefore, removed; and (4) the transmission channel medium, which can phase distort the speech waveform, resulting in a significantly distorted signal from which to extract the pitch period.

2.3.1 Comparative Performance

Four of the seven methods evaluated include: (1) The Real Time Pitch Detector by Dubnowski *et.al.,* using centre-clipped and infinite-clipped speech signal, prior to an Auto Correlation Function [13]; (2) the Log-Frequency Domain Cepstral method by Noll [6]; (3) the Simplified Inverse Filter Tracking (SIFT) algorithm by Markel [16]; and (4) the Average Magnitude Difference Function (AMDF), by Ross *et.al* [17]. The above algorithms performed well in the category of Fine Pitch errors (error not exceeding 1ms), with the exception of the AMDF. The limitation of micro-processor instruction speeds in 1976 forced the AMDF signal to be down-sampled (reducing the instruction count) to attain real-time

capability, which subsequently resulted in a reduced AMDF resolution [15]. The improvement of micro-processor instruction cycle time has removed this limitation. The AMDF, if implemented on current hardware (with higher instruction speed), would be expected to show improved performance. This improvement can be achieved as a result of the increased resolution in the final AMDF output. The recent (1992) use of the AMDF in [18] supports this. The evaluation in [15] concluded that no algorithm was superior.

Evaluations, performed by Markel and Rabiner *et al.*, comparing the performance of the SIFT algorithm to that of the Cepstrum, have independently concluded the SIFT algorithm improved accuracy [15][16]. The Cepstrum technique is the "standard by which other approaches are measured" [19], [26], but a reported disadvantage is the necessity to utilise the full speech bandwidth to attain a large number of harmonics [15]. The SIFT algorithm is therefore implemented in this thesis, and the results are presented.

2.3.2 The Average Magnitude Difference Function

In the AMDF the summation of the differences between a segment of voiced speech and that of a delayed version of the segment is calculated as:

$$AMDF(d) = \frac{1}{k} \sum_{n=q}^{q+K-1} \| s(n) - s(n+d) \| \quad \text{.....(2.6)}$$

where *s(n)* is the speech signal and '*d*' is the delay index.

Modifications to this distance function have included the use of the Linear Prediction (LP) residual, *r(n)*, in lieu of the raw speech signal, to improve performance. LP was used because the speech signal is sensitive to intensity variations, noise and low frequency spurious signals [7]. In this latter case the AMDF is represented as:

$$AMDF(d) = \frac{1}{E} \sum_{n=q}^{q+K-1} \| r(n) - r(n+d) \| \quad \text{.....(2.7)}$$

where $d = 1 .. K$, and $E = \sum e(n)^2$ is the energy of the residual *r(n)*.

The AMDF, computed at zero delay, evaluates to zero. As the delay index increases, the resulting AMDF outputs reduced values (approaching zero) at delays equal to the pitch period. The AMDF's [16] performance in detecting the pitch period improves as a result of the pronounced signal at glottal epochs. This is in contrast to the performance of the ACF technique. In demonstrating this improved performance, the AMDF has been expressed in terms of the square of the ACF, offset by the DC term R(0). The Root-Mean-Square (RMS) of the difference (error) signal is used in addition to the short-time stationary assumption of the speech signal. When the ACF is large compared to the value of R(0) the AMDF will approach zero. The recorded distance between these nulls is a direct measure of the pitch period. When encountering the onset of a voiced segment within a frame that contains a mixture of unvoiced and voiced speech, a reversal of the time function is suggested in [17], to enable rapid pitch detection. This time reversal allows the voiced portion to be included in the AMDF-integrating window [17].

2.3.3 Evaluation Criteria

The four major error categories defined in [15] are (1) Gross Pitch Error, where the calculated pitch period exceeded (+/-)1ms with respect to the reference pitch period; (2) Fine Pitch Error is defined as an error less than or equal to (+/-)1ms (excluding no error). The mean and standard deviation were also calculated for this category; (3) Voiced-to-Unvoiced errors, defined as misclassifications of voiced speech frames, and (4) Unvoiced-to-Voiced errors, defined as misclassification of unvoiced speech frames. Categories (1) and (2) will be used in the evaluation of pitch detectors presented in this thesis. These will provide the accuracy measure of the final pitch contour.

In addition, the following criteria will be used to analyse algorithm performance: (1) the algorithmic delay, (2) the extraction rate of the pitch period, and (3) the robustness within poor signal environments. The performance of the algorithms with or without the presence of specific frequency components will also be discussed in this Chapter.

Voiced/Unvoiced (V/UV) classification is not a specific objective of this thesis. Classification of V/UV frames will be reported if the Pitch Detection technique incorporates such a classification, otherwise no objective evaluation is performed. A selection of V/UV and extensions into V/UV/Mixed and Silence algorithms are presented in [20], [21], [22], [23].

2.4 Correlation based Pitch Detection

2.4.1 Auto Correlation based Pitch Detection

The Auto Correlation Function (ACF) is regarded by Rabiner [24] as one of the most successful pitch detection methods. The ACF is a power spectrum, and is, therefore, phase insensitive. This is an advantage when considering a signal that has been phase-distorted by the transmission channel. The presence of secondary pitch candidates (Formant frequencies) impedes accurate pitch determination, but with Linear Prediction and subsequent Inverse Linear Predictive filtering, a significant elimination (whitening) of the Formant structure can be achieved. However, lower Formant frequencies will still be present [24], and can be detrimental to Pitch Detection. The application of non-linear techniques has demonstrated (Section 2.2) an improved performance over the ACF based techniques that operate on the raw speech signal alone for the removal of secondary pitch candidates.

A number of variants combining the centre-clipping and infinite-clipping functions were investigated by Rabiner [24], who concluded that the "differences in the performance of the remaining correlators (non-linearly processed) were not consistent. Thus, any one of these correlators would be appropriate". The degree to which secondary components influence accurate pitch detection, and the techniques used to remove such components in ACF detectors, is illustrated by the above examples.

Reductions in ACF computational complexity (to achieve real-time processing) are mostly concerned with removing redundant calculations. These reductions are achieved by

considering values of delay that are most likely to yield pitch estimates that generate smooth pitch tracks. In these circumstances, the ACF is calculated in the vicinity of the previous delay value. Consequently, it is important, that methods which implement this scheme generate accurate pitch estimates in order to avoid incorrect tracking. Additionally, accurate initial estimates are required when making a transition from an Unvoiced-to-Voiced speech segment. The use of adaptive windows in the computation of the ACF, to account for the pitch variation, is appropriate given the time-variant characteristic of the speech signal [24]. The disadvantage of these windows, however, is the delay incurred when final decisions are subjected to error-correcting procedures. Speech characteristics, such as pitch doubling or tripling are less pronounced due to the ACF inherently tapering to zero, placing a bias towards the lower pitch period values [13].

2.4.2 Maximum Likelihood and Auto Correlation

The Maximum Likelihood (ML) Pitch Estimation technique, by Wise *et al.*, and the Pseudo Maximum Likelihood (PML) method proposed by Freidman, which were published within a short period of each other, provide a basis for ML-based pitch detection [25][26]. The ML approach is based on the minimisation of the squared differences summed between a period estimate of the speech signal, and a periodic repetition of the estimate segment. The PML pitch period is the delay value which maximises the weighted energy of a periodic signal, $s_O(t)$, defined as the weighted average of periods (delays) of the candidate periodic signal, as given by equation (2.8).

$$I_0(t_0) = \sum_n \int_0^T \frac{s(t)w(t)s(t+n.t_o)w(t+n.t_o)}{\sum_k w(t+k.t_o)}.dt \quad \text{.....(2.8)}$$

where $w(t)$ is the Hanning window function applied to the speech signal $s(t)$, and $n.t_0$ is the delay function.

Two important aspects that arise from the PML formulation are: (1) use is made of the signal periodicity. This is due to the determination of the minimum weighted mean square error, assuming a periodic signal and stationary signal over short time intervals; and (2) the PML solution is not dependent on the presence of "specific spectral components, eg., the fundamental" [26]. Pitch detection methods that possess this capability are advantageous over methods which rely on the existence of the Fundamental component.

It has been suggested by Friedman [27] that utilising the LP residual for the PML technique may improve results. The ML and PML techniques have shown, independently, that a ML solution inherently calculates the Auto Correlation Function [26],[27]. An observation of the PML estimate function (2.8) reveals the summation of a series of autocorrelations for a candidate signal segment, $s(t)$ [26], [27]. This provides a strong basis for the continual use of the time-domain based methods, such as autocorrelation based techniques. Low-Pass filtering may remove information pertinent to accurate pitch detection. Consequently a"pitch estimate is based on the periodic structure of the speech signal as a whole, and not mainly on the Fundamental and lower harmonics" [7].

2.4.3 Cross Correlation Pitch Determination

The Cross Correlation Function (CCF) is computed from two adjacent speech segments which may overlap [6]. In contrast, in the Short-Time Auto Correlation Function (ACF) method [6], only the speech samples within the selected segment are used. Commencing from the zero'th autocorrelation value (where all values are utilised) the number of terms reduces as the index increases, and hence, the ACF tapers to zero. The CCF, conversely, does not taper as the signal samples are made available, either at the beginning or end of the speech segment. This results in a uniform correlation function. These underlying mechanisms compare with those used in Linear Prediction Analysis [6]. The "Super Resolution Pitch Determination of Speech Signals" [11] presents a Pitch Detector which is based on the CCF. The method is based on the minimisation of the Normalised Squared Error (NSE) criteria. The CCF is calculated for the full range of pitch periods, from which the optimal candidate is selected.

Selecting a CCF global maxima does not necessarily provide the correct pitch estimate. As with the Auto Correlation Function technique, the CCF is computed in the vicinity of the previous pitch estimate. The aim, once again, is to reduce correlation computations, and generate smooth pitch tracks. The pitch tracking highlights the consequence of selecting a global maximum, and the ambiguity that exists when multiple CCF candidate peaks are present. To avoid pitch doubles or triples, tests for the presence of lower harmonics were undertaken [11] to select the optimal pitch period. CCF thresholds assist in the generation of smooth pitch tracks, as do ACF thresholds.

2.5 Attaining Lower Bit-Rate Pitch Parameters

In [28] an historical review, detailing the important developments which have had the greatest impact in speech coding, is presented. The review traces the evolution from Linear Prediction-based Analysis-by-Synthesis (A-by-S) Speech Coders (LPAS) to Code-Excited Linear Prediction (CELP). Further developments of CELP include the introduction of adaptive code books. The improvements in speech quality obtained from CELP were a direct result of the advances in Long-Term (Pitch) Prediction. The resultant higher pitch period resolution, however, incurred a bit-rate increase for the pitch component. Subsequent interpolation of the pitch profile negated this increase, and the implementation of complex fractional pitch determination techniques was no longer considered a necessity [28][29][30].

The CELP adaptive code-book index (delay) is interpolated in order to reduce the pitch component bit-rate [31]. An incorrect delay estimate will result in a time mismatch in the reconstructed Linear Prediction (LP) residual. A time alignment or time-warp (non-linear mapping) of the delay is undertaken in CELP. The LTP parameters then become deterministic functions of time which allow them to be parameterized, permitting linear interpolation of the delay contour. Adaptive code books which are populated with various time-warped LP residual vectors are used in lieu of adaptive code-books comprising raw LP residual vectors. In Analysis-by-Synthesis (A-by-S) the input (speech signals) to the analysis phase is modified so that synthesis is simplified to produce what are perceived to be equivalent results [31]. Removing redundant information permits interpolation techniques to be used without fear of degradation in the reconstructed signal. This strategy models the pitch contour by modifying the original speech signal so that it can be represented by a piece-wise linear trajectory.

The process involves sub-dividing the speech frame into sub-frames whose lengths are ideally adapted to the Pitch Period [31]. The difficulty in determining correct estimates for the LTP sub-frame boundaries has led to the implementation of peak-picking algorithms, which locate high energy regions (spikes) in the Linear Prediction (LP) residual. To maintain a continuous delay contour, the location of sub-frame boundaries needs to be chosen carefully. The use of pitch pulses was regarded in [31] as a satisfactory method in locating sub-frame boundaries.

The development of the 16 kbit/s Low Delay (LD) - CELP by Chen *et al.* [9] which was subsequently adopted by the ITU-T in 1992 (G.728), eliminated explicit pitch determination. In [9] the algorithm removed explicit pitch-prediction and replaced it with a 50th order Linear Prediction filter. Due to the poor robustness and unacceptable algorithmic delay of candidate pitch detection algorithms, they were considered unsuitable. Forward and backward pitch prediction was trialed in [9], with little success, concluding that the forward predictor exceeded the delay constraint, and that the backward predictor did not provide sufficient robustness.

Current research into the development of low bit-rate speech coding encompasses both well-founded Vocoding techniques, which have been revisited, and recent Waveform Coding Techniques [28]. The Prototype Waveform Interpolation (PWI) technique is considered [32], to be an amalgamation of both these two these techniques. At lower bit-rates (2400bps), the Prototype Waveform Interpolation and Time-Frequency Interpolation methods offer substantial improvement over LD-CELP. However the loss of naturalness and a certain noticeable false periodicity (Buzziness) is reported. In addition to this, a significant coding delay is incurred.

In addition to the Pitch Detection methods that have been discussed in this Chapter, two of the most promising Pitch Detection algorithms are implemented next in this Chapter with their resultant pitch profiles. The Simplified Inverse Tracking Algorithm (SIFT) [16] and the Glottal Closure Interval (GCI) [33] methods are investigated as a baseline for comparison with the new techniques to be presented in Chapters 3 and 4. The SIFT algorithm is based on the partial removal of unwanted Formant frequencies which pseudo-whitens the speech signal. Significant signal periodicities that contribute to the Fundamental component are retained as a result of the pseudo-whitening process [16]. In the GCI technique the impulse response of the vocal tract and the instance of glottal opening are combined and crosscorrelated with the speech signal (residual). This resulting signal is used to enhance the instances pitch epochs.

2.6 Simplified Inverse Filter Tracking Algorithm

2.6.1 SIFT Introduction

The Simplified Inverse Filter Tracking (SIFT) Algorithm is based on a reduced-order Linear Prediction (LP) Filter, and is similar to that used for the Vocal Tract Filter [16]. The ACF is implicitly used in calculating the LP coefficients, and also in the determination of the final pitch period estimate. The reduced order LP filter is the reason for the algorithms simplicity and, in addition, the down-sampling operation significantly reduces the number of ACF computations. The strengths [16] of the SIFT algorithm in detecting voiced plosives and accurate pitch period generation are considered important attributes for low-delay pitch detection. The SIFT algorithm performed successfully, however, the algorithm was not considered to be robust, as certain entire voiced speech segments failed to be detected correctly (even when including tests for pitch period multiples or triples).

2.6.2 The SIFT Algorithm

The major components of the SIFT algorithm are illustrated in the block diagram of Figure 2.7. The input speech is initially Low-Pass Filtered (LPF) with a cut-off frequency of 0.9kHz. The resulting filtered speech is then down-sampled (D↓4:1) prior to determining the 4th Order Linear Prediction (LP) coefficients. The speech signal is then Inverse-filtered by the LP filter, removing formant frequency components sufficiently, but not entirely, thereby retaining significant pitch information. This short filter length results in a pseudo-whitening filter emphasising the pitch periodicities while removing higher formant frequency components.

Simplified Inverse Filter Tracking (SIFT) Pitch Detector

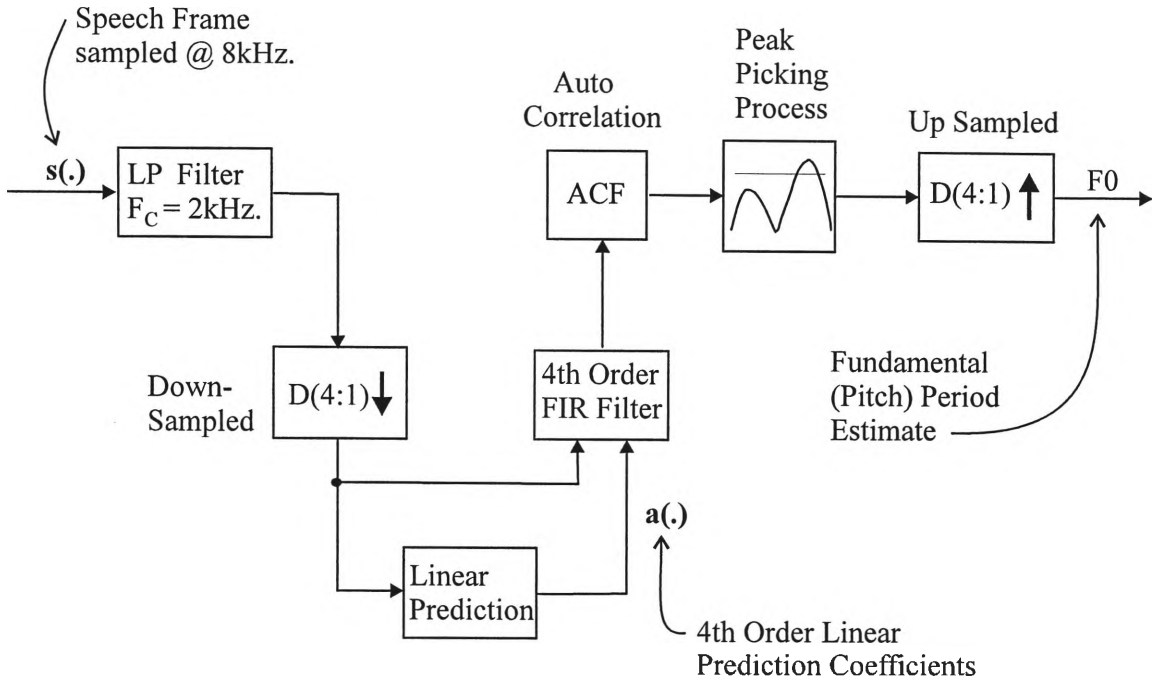

Figure 2.7 - SIFT Pitch Detector Block Diagram.

A reduced ACF is subsequently performed on the Inverse filter output. Prior to determining the final pitch estimate, an ACF threshold of 40% with respect to the frame energy, assists to discriminate between voiced and unvoiced frames [16]. The V/UV decision algorithm [16] prevents anomalies when adjacent preceding and succeeding frames exceed the threshold, but the current frame does not. Once the frame is classified as voiced the maximum ACF position is determined, and interpolation is performed to return to the original resolution prior to the down-sampling operation [16].

2.6.3 The SIFT Algorithm Implementation

2.6.3.1 Reduced Autocorrelation Function

An attribute of the SIFT algorithm is the inherent tracking of the autocorrelation peaks on a frame-by-frame basis. This is achieved by including accurate (previous and initial) pitch period estimates in a reduced ACF. A window of 16 samples (+/- 4ms) wide, surrounding the current auto-correlation peak, is considered the greatest area of interest in tracking the pitch period variation. This corresponds with [16], where it was considered an unnecessary computational cost to evaluate all possible ACF values. The slow rate-of-change of the pitch variation does not warrant an extensive search.

2.6.3.2 Algorithmic Delay

One of the disadvantages with the SIFT algorithm is the relatively large delay incurred prior to final pitch determination. A delay of two frames, to allow for corrections to the final pitch estimate prior to transmission (resulting from peak-picking errors), is incurred. This large delay is considered unsuitable for use in Low-Delay pitch detection today even though in 1974 the emphasis was purely on low bit-rate speech coding [34].

2.6.3.3 Fundamental (F0) Presence

The inability of the SIFT algorithm to discriminate between the first Formant frequency (F1) and that of the Fundamental (F0), when both are present within a narrow frequency band, is another disadvantage. In some instances the Fundamental is not present, as it is either

removed by the Formant filter, or the signal itself does not contain a dominant Fundamental frequency component. The absence of a dominant Fundamental component for certain high pitch speakers is also attributed to high-pass filtering (0-300Hz cutoff) incurred by the channel transmission medium. Thus, algorithms such as the SIFT, which rely on the presence of the Fundamental, tend to fail in these circumstances. Two speech segments which illustrate the removal of the Fundamental component by the Vocal Tract (Formant) Filter are illustrated in Figures 2.8a-b and 2.9a-b. Figure 2.8b clearly illustrates a low amplitude across the entire residual frame, even though a repetitive component may be observed in the speech signal (Figure 2.8a). In Figure 2.9b no indication of the pitch period is observed in the residual segment. In addition, if the speech in Figure 2.9a is high-pass filtered, the sinusoid component will be removed from linear prediction analysis.

A solution to the F0/F1 discrimination problem utilises a COMB filter [7] to replace the LPC filter. An adaptable cut-off frequency ensures that the F0 component is present within the filter pass-band of the COMB filter. The F0 is subsequently assumed to be present in the filtered output prior to an ACF calculation [7][19].

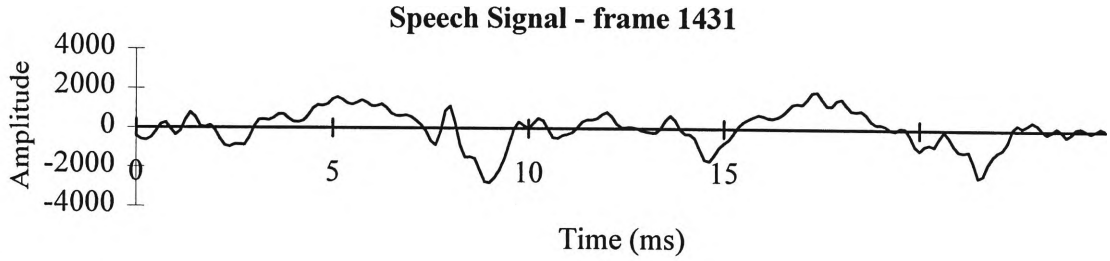

Figure 2.8a - An example of a low pitch speech frame.

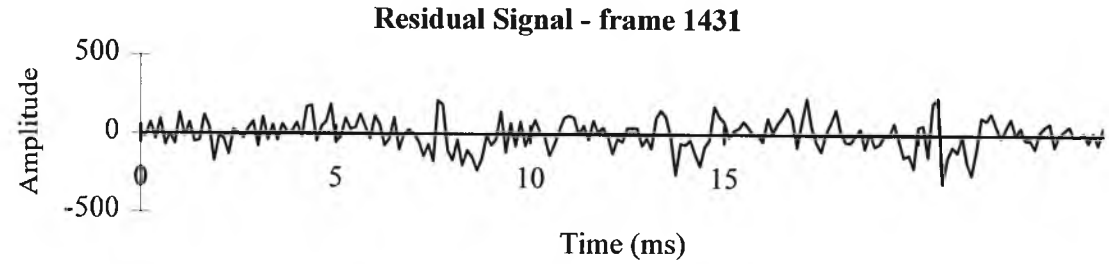

Figure 2.8b - The Fundamental component is removed from a low pitch segment by the Formant filter, resulting in a low amplitude across the residual frame.

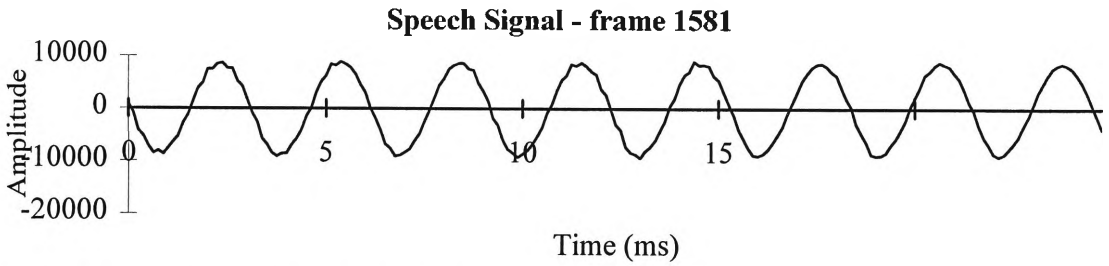

Figure 2.9a - An example of a high pitch speech segment.

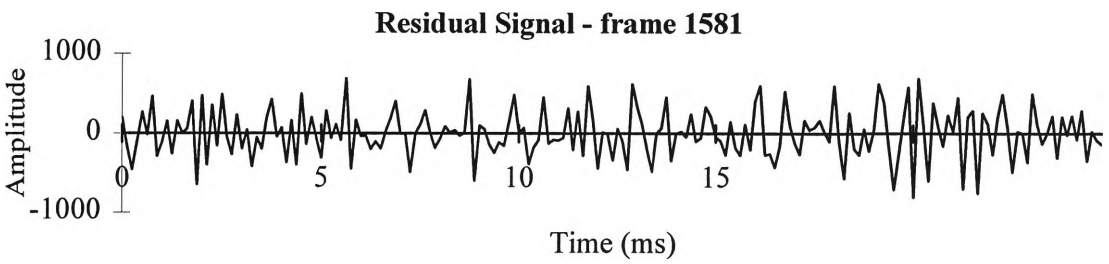

Figure 2.9b - The Fundamental component is removed from a high pitch segment by the Formant filter (no periodic component is observed in the residual segment).

2.6.4 SIFT Algorithm Pitch Detector Results

The SIFT Algorithm Pitch Detector was executed using a twenty sentence speech database (2024 frames) of 'British' English language speakers [35]. The results of the SIFT algorithm are presented in Table 2.1, with sample pitch profiles generated by the SIFT algorithm illustrated in Figures 2.10a-b. A complete set of the result profiles are presented in Appendix A. The Table and Figures illustrate accurate results for a majority of speech segments, however, the algorithm has failed to accurately track certain voiced segments. It has been observed (for example in Appendix Figure A8) that in higher pitch period segments associated with the male speakers, the algorithm has failed to correctly discriminate between the true pitch period and a harmonic of the Fundamental, resulting in a lower pitch period being recorded. This confirms the deficiency reported by other authors [7].

TABLE 2.1

Results for the SIFT Pitch Detector applied to a 2024 frame speech database of which 1079 are voiced.

Error Classification	Count	Average (samples)	Std. Dev. (samples)
Voiced frames with no Errors	259	0	N/A
Voiced frames with Fine Errors [≤ 8 samples (≤ 1ms)]	725	2	1
Voiced frames with Gross Errors [> 8 samples (> 1ms)]	95	32	21
Total Voiced frames	1079	4	12

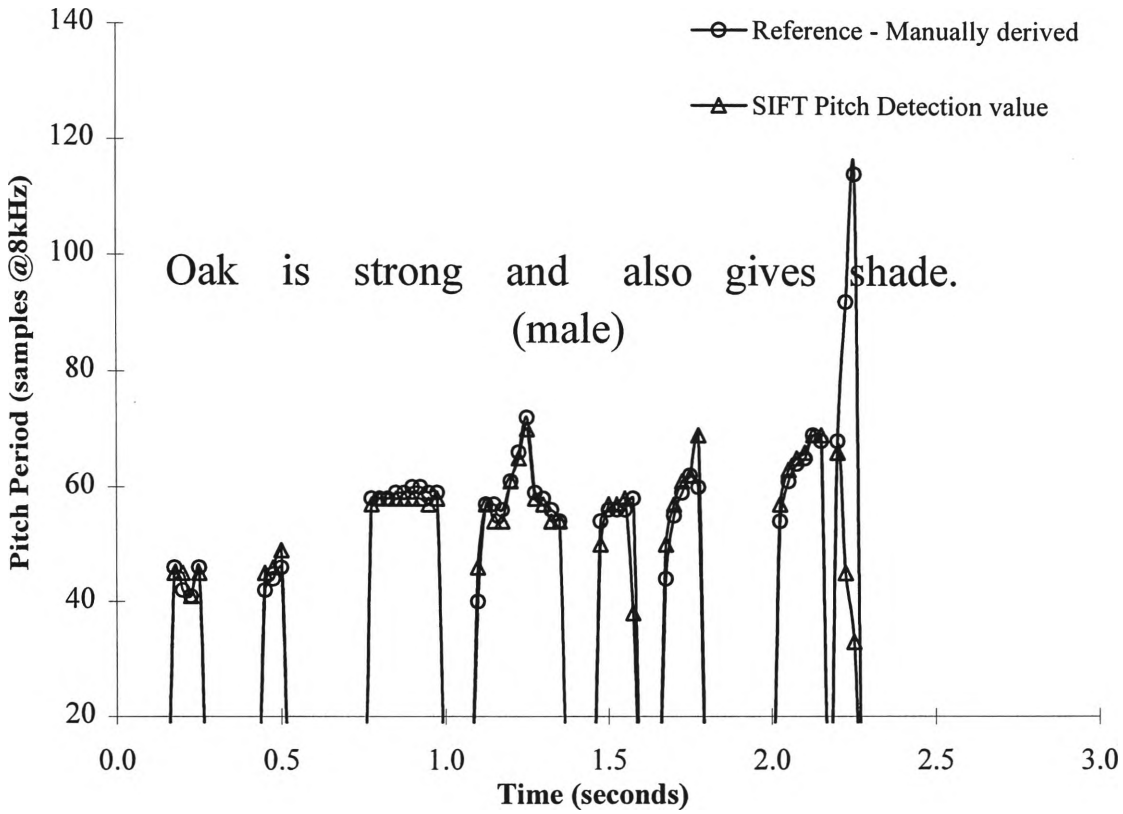

Figure 2.10a - SIFT Pitch Detector resultant pitch profile.

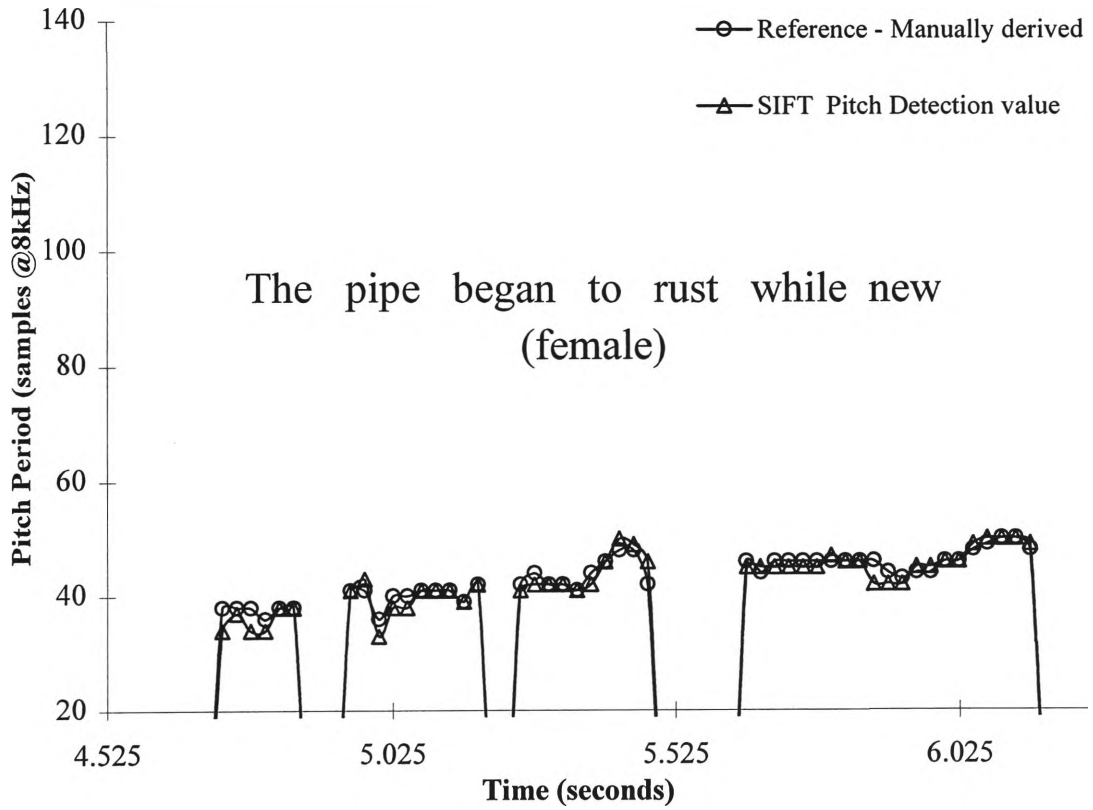

Figure 2.10b - SIFT Pitch Detector resultant pitch profile.

2.7 Glottal Closure Interval Pitch Detection

2.7.1 GCI Introduction

The Glottal Closure Interval (GCI), as a measure in determining the pitch period, has received considerable attention [33][36][37]. Pitch Detection methods which use the GCI measure the distance between two consecutive glottal closure instances. It is the opening of the glottis which allows the passage of air, stored within the lungs, to pass through the vocal cords, causing them to vibrate and generate the desired sound. The point in time at which vocal cords separate constitutes the Glottis opening, and is therefore the source of pitch information [7]. The benefits gained by detecting the instances of glottal openings and the period between glottal openings provides the algorithm [33] with the capabilities to detect voiced-to-unvoiced transitions, and to track period-to-period variations. These important attributes have warranted further investigation in this thesis, as it is considered a viable algorithm for high quality, low-delay pitch detection [33]. The GCI Pitch Detector was implemented and the corresponding results are presented.

The basis of the technique is the enhancement of the instances of glottal closures. In achieving this result, two principle processes are: (1) the crosscorrelation of the speech signal with the impulse response of the vocal tract, yielding a Maximum Likelihood (ML) signal; and (2) the further enhancement of the ML signal by the corresponding Hilbert Transformation (HT) envelope. In enhancing the glottal periodicities, the HT envelope is superimposed onto the ML signal. This emphasises the GCI instances while de-emphasising secondary pulses. The HT envelope of the ML signal meets the criteria for an appropriate "Selection Signal" in [33].

A block diagram illustrating the sub-systems of the GCI Pitch Detector is presented in Figure 2.11.

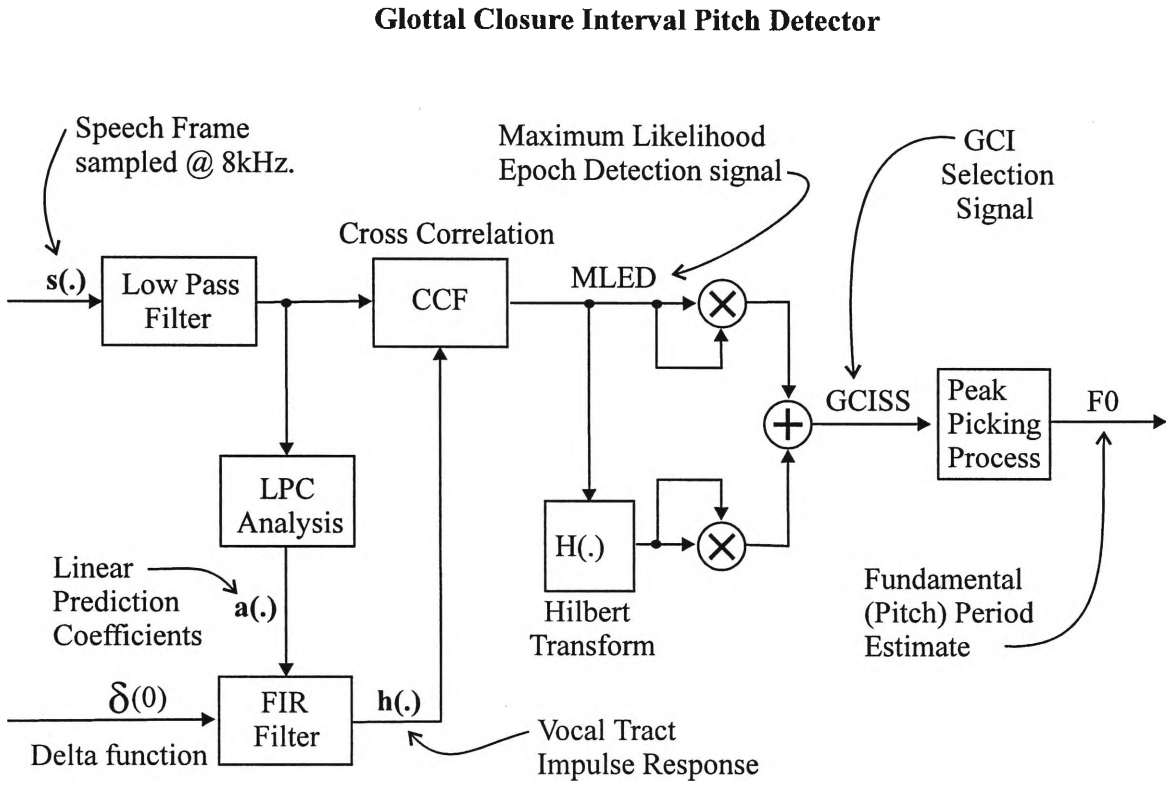

Figure 2.11 - Glottal Closure Interval Pitch Detector Block Diagram.

2.7.2 Determining the Instant of Glottal Closure

The "Automatic and Reliable Estimation of the Glottal Instant and Period" [33] is based on modelling the Glottal Closure Interval (GCI) by an appropriate impulse response, $h_{g.c.}(n)$. This impulse response comprises a delta function $\delta(0)$ *at* $n = 0$ and the impulse response of the vocal tract, $h_{v.t.}(n)$, *for* $n > 0$, (2.9). The use of the glottal closure impulse response results in a more reliable detection of the glottal discontinuity over methods that rely on high

energy regions in the Linear Prediction residual. Such high energy regions may not be present, as shown previously in Figure 2.4b. Additionally, the occurrence of multiple pulses causes an ambiguity in selecting the correct GCI pulse (an example is illustrated in Figure 2.5d).

$$h_{g.c.}(n)=\left\{\begin{array}{ll}=0; & n<0\\ =\delta; & n=0\\ =h_{v.t.}(n); & 0<n<40ms\end{array}\right\} \quad \text{..... (2.9)}$$

2.7.2.1 Maximum Likelihood Epoch Detection Signal

The Maximum Likelihood (ML) function is the Cross Correlation Function (CCF) calculated from the impulse response $h_{v.t.}(n)$, and the speech or, residual signal. This CCF results in a ML-Epoch Detection (MLED) signal [36]. In [33] it is assumed that the difference (error) function between the observed speech and that of the reconstructed speech signal is Gaussian. ML techniques are, therefore, permitted, and it is shown in [33] that the dominant term influencing the maximum-likelihood estimation function is that given by (2.10).

$$\sum_{n=0}^{N-1}\frac{s(n+n_0)\,\hat{s}(n)}{2\sigma} \quad \text{..... (2.10)}$$

where $\hat{s}$ is the reconstructed (linear predicted) speech signal, $s(n + n_0)$ is the delayed speech signal, and σ is the variance of the difference signal. The summation occurs for the length of impulse response N.

2.7.2.2 Glottal Closure Instant Selection Signal

The Glottal Closure Instant Selection Signal (GCISS) [33], is introduced to enhance the signal periodicity of the waveform, to the detriment of spurious signal noise, and to dampen dominant secondary pitch candidates. The GCISS, as defined in [33], is a periodic pulse train, with a certain pulse width to accommodate for the variation in pitch period. Observations from candidate selection signals, such as rectangular, triangular, and a Gaussian pulse train, have revealed a common set of properties [33]. These properties are: (a) the spectrum of the selection signal is symmetric and real; and (b) the spectrum has its maximum magnitude at the origin with a gradual roll-off in amplitude. The Hilbert Transformation envelope of the MLED signal is demonstrated in [33] to satisfy this requirement. Derived from the MLED, the GCISS results in a waveform where the instances of glottal closures are enhanced, and is defined as:

$$GCISS(n_0) = \sqrt{\left[MLED^2(n_0) + H\{MLED(n_0)\}^2 \right]} \quad \text{..... (2.11)}$$

2.7.3 Benefits of the Glottal Closure Interval Selection Signal

The performance degradations of previous GCI-based algorithms, as reported in [36], have been attributed to the ambiguity in locating the correct epoch position when using the Linear Prediction (LP) residual. LP analysis assumes an all-pole model which implicitly implies a minimum-phase characteristic. If this assumption fails, as it does for nasal-couplings, the speech signal cannot be accurately reconstructed with respect to the phase response.

In [36][37] the Hilbert Transformation was successfully applied to the residual, concluding that ambiguities pertaining to the exact location of the glottal openings can be removed. The use of the Linear Prediction (LP) residual, however, exposed its own deficiencies in that epochs for certain voiced segments do not exist. The LP residual does not contain the distinctive epochs which define the closed glottis interval due, either to LP, or High-Pass channel filtering. The residual for these segments will be difficult to process by methods which are dependant on residual epochs. Signal periodicity, which is stressed by Cheng & O'Shaughnessey, [33] is used in deriving an appropriate GCI Selection Signal (GCISS). By selecting the appropriate GCISS, the enhancement of the GCI epoch occurs, and the influence of secondary pulses is diminished [33]. The use of the GCISS removes the ambiguity associated with the presence of sub-pulses in the MLED signal.

The GCI method [33] included a logical Voiced/Unvoiced/Mixed decision process to classify the speech segment accordingly. This decision process included heuristics common to Pitch Detectors similar to those presented in [20][38]. If the distance measured between consecutive pulses in [33] was not valid (off-track), then the analysis frame was classified unvoiced. A valid estimate of the pitch is that value which results in a smooth and continuous pitch track. The GCI Pitch Detector in this thesis did not implement such a V/UV/M decision process because a manually derived pitch profile was used as the reference.

2.7.4 GCI Pitch Detector Implementation

The GCI Pitch Detector presented was implemented as outlined in [33]. Modifications made to maintain a common reference with other Pitch Detection methods presented in this thesis

included: (1) the use of 200 sample analysis frames (speech sampled @8kHz); and (2) the use of a 10th Order Linear Prediction (LP) filter in lieu of the 12th Order LP filter, as used in [33]. The Hilbert Transformation Filter was implemented in the time domain using the coefficients provided in [33].

2.7.5 GCI Pitch Detector Performance

Figures 2.12a-g illustrate the algorithm's performance at the onset of voiced speech. The speech sample represents the commencement of the letter 'a' in the word '**a**nd'. The enhancement of the glottal (pitch) pulses is clearly evident, however, due to the solitary pulse at the end of frame 43, no pitch value was recorded against this frame. In the current implementation, the final pitch estimate is computed on a frame-by-frame basis. The algorithm has successfully located the epoch occurrence at the onset of this voiced speech segment as shown in Figure 2.12d, and the following three glottal intervals in Figure 2.12g.

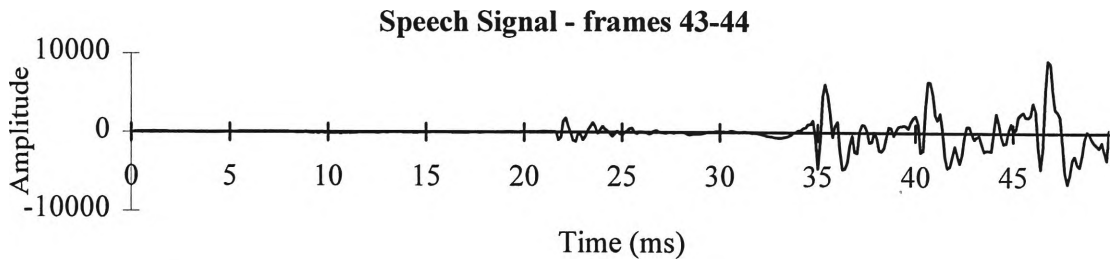

Figure 2.12a - Contiguous speech frames 43 and 44 (spanning 50 ms) that illustrate the onset of voiced speech.

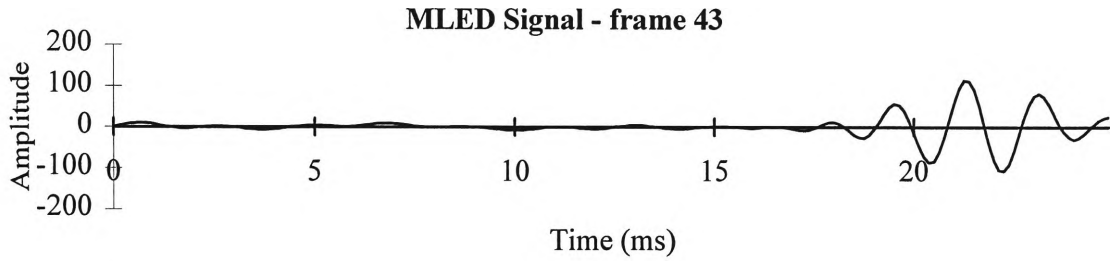

Figure 2.12b - Maximum Likelihood Epoch Detection Signal for speech frame 43.

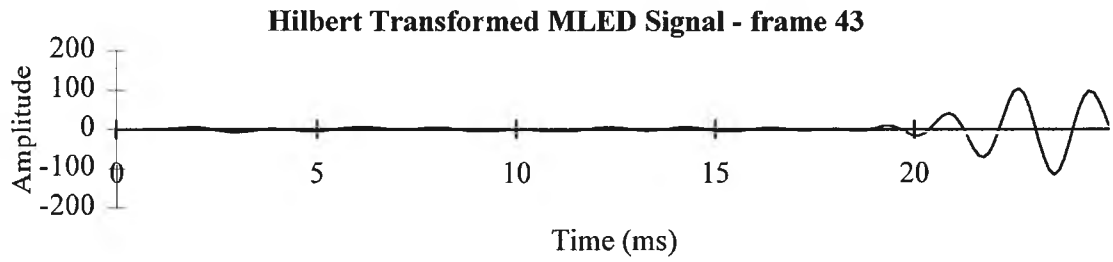

Figure 2.12c - Hilbert Transformed MLED Signal for speech frame 43.

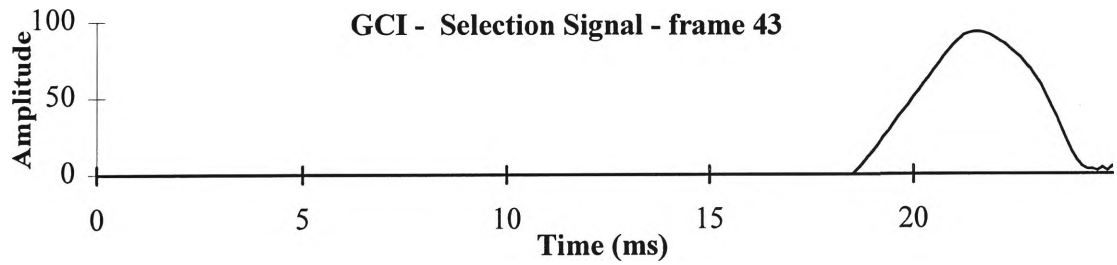

Figure 2.12d - Glottal Closure Interval Selection Signal for frame 43. The solitary pulse recorded for this frame indicates the onset of a voiced speech frame.

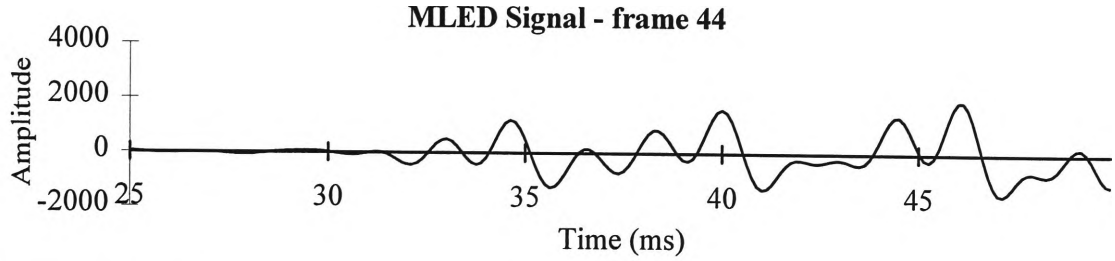

Figure 2.12e - MLED Signal for speech frame 44.

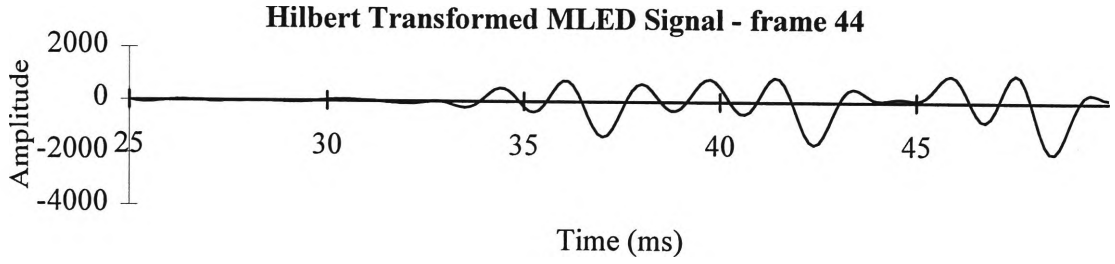

Figure 2.12f - Hilbert Transformed MLED Signal for frame 44.

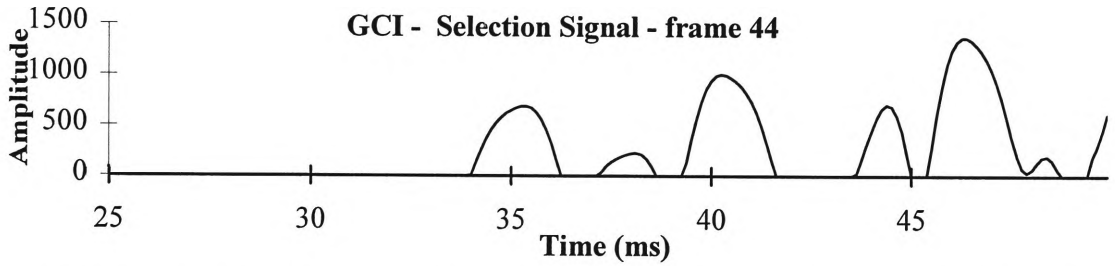

Figure 2.12g - Glottal Closure Interval Selection Signal for frame 44. The reduced influence of the secondary pulses is evident. The three largest peaks are a direct measure of the glottal closure interval.

Figures 2.13 - 2.14, contiguous speech frames (645-646), illustrate the output signals at various stages of the GCI determination process. The final pitch period estimate is extracted from the Glottal Closure Instant Selection Signal (GCISS) via a peak-picking algorithm. From these figures it is observed that the GCI technique has correctly determined the closed glottal interval.

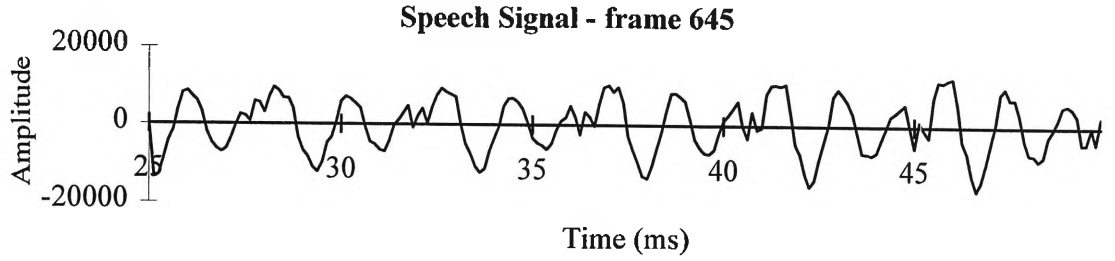

Figure 2.13a - Voiced speech frame 645.

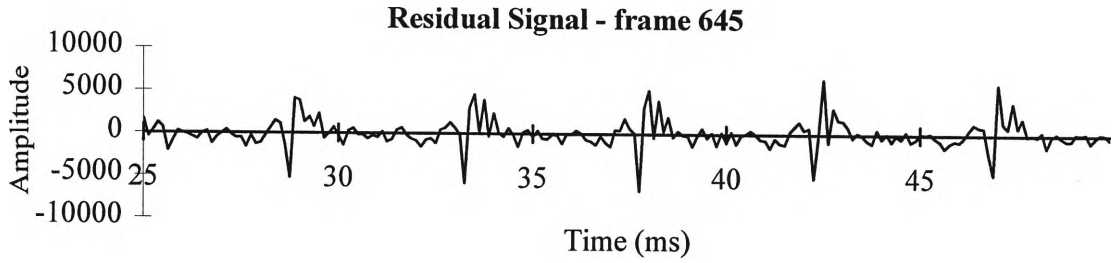

Figure 2.13b - Residual frame illustrating the distinctive glottal epochs.

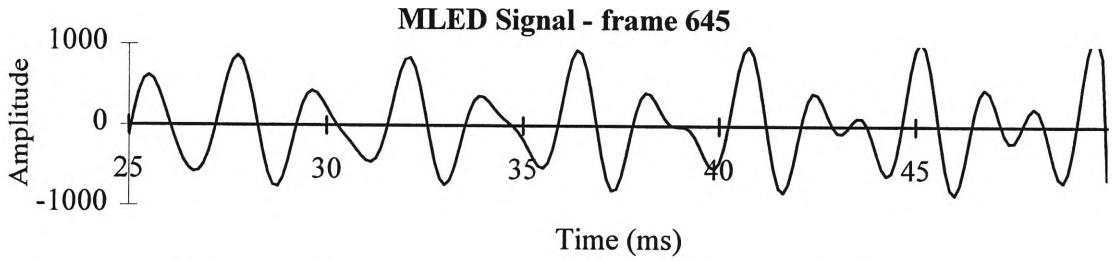

Figure 2.13c - Maximum Likelihood Epoch Detection Signal (MLED).

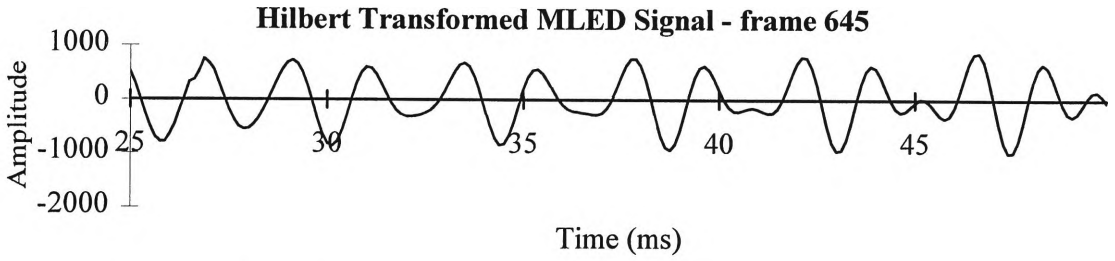

Figure 2.13d - The Hilbert Transformed MLED signal.

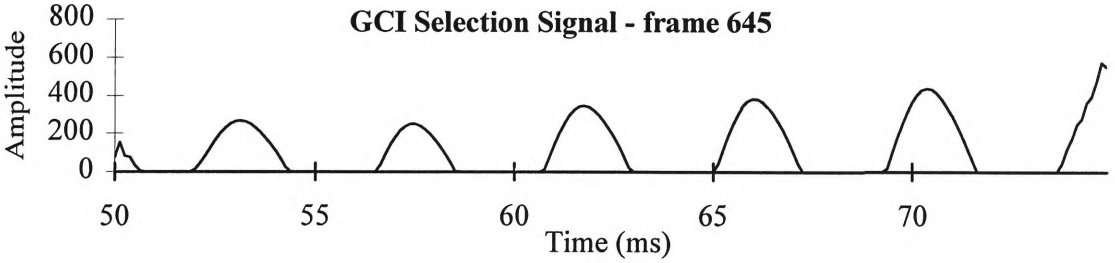

Figure 2.13e - GCI Selection Signal for frame 645.

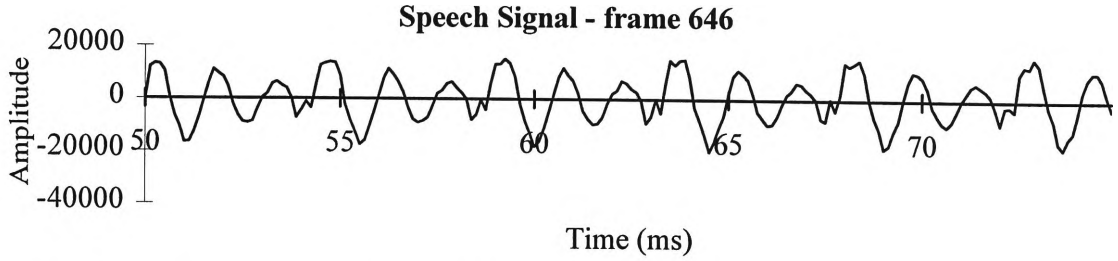

Figure 2.14a - Voiced speech frame 646.

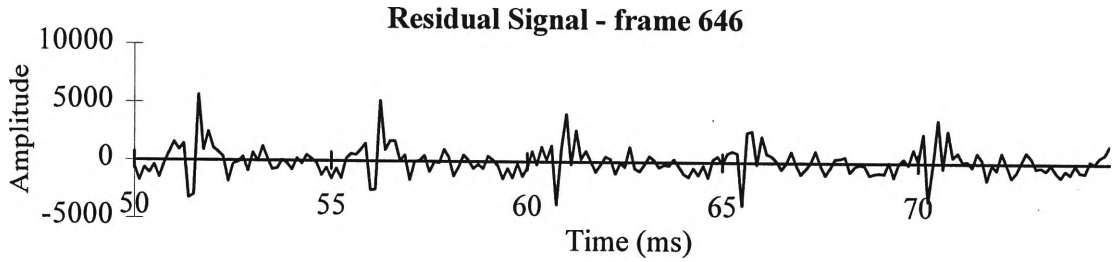

Figure 2.14b - Residual frame illustrating the distinctive glottal epochs.

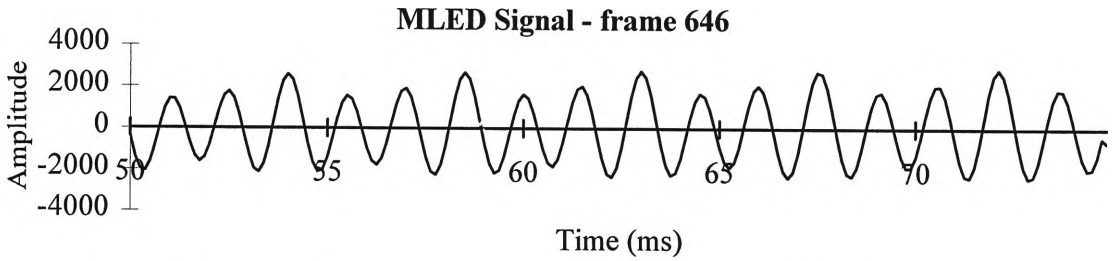

Figure 2.14c - Maximum Likelihood Epoch Detection Signal (MLED).

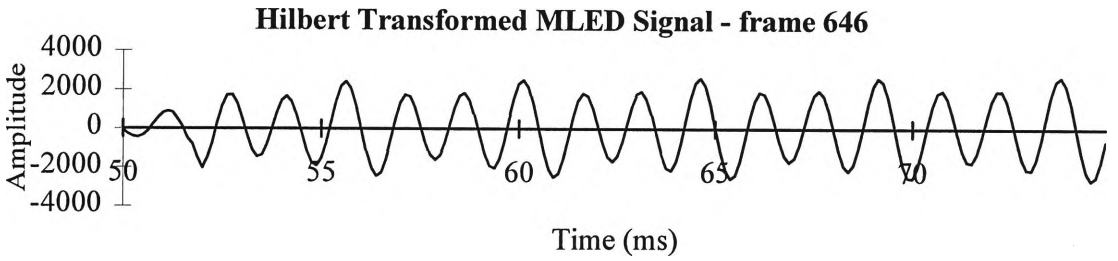

Figure 2.14d - The Hilbert Transformed MLED signal.

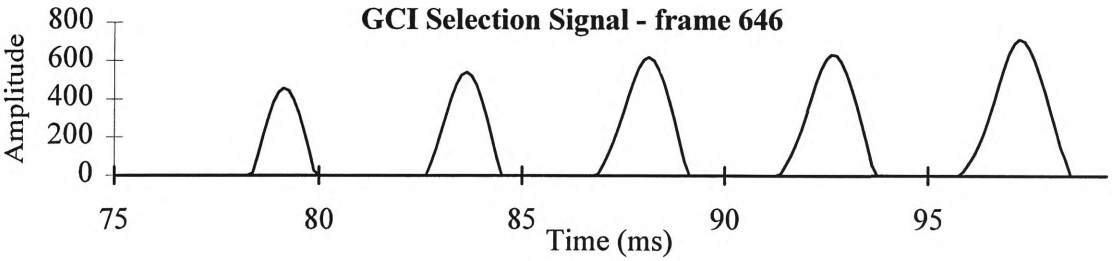

Figure 2.14e - GCI Selection Signal for frame 646.

2.7.6 GCI Pitch Detector Results

The GCI Pitch Detector generated results using the same 20 sentence (2024 frames) speech database [35] as that used for the SIFT Pitch Detector. The results of the GCI Pitch Detector are presented in Table 2.2, with sample pitch profiles generated by the GCI algorithm illustrated in Figures 2.15a-b, and the complete set of the result profiles are presented in Appendix B.

TABLE 2.2

Results for the GCI Pitch Detector applied to a 2024 frame speech data base, of which 1079 are voiced.

Error Classification	Count	Average (samples)	Std. Dev. (samples)
Voiced frames with no Errors	239	0	N/A
Voiced frames with Fine Errors [≤ 8 samples (≤1ms)]	691	2	2
Voiced frames with Gross Errors [> 8 samples (>1ms)]	149	28	21
Total Voiced frames	1079	5	14

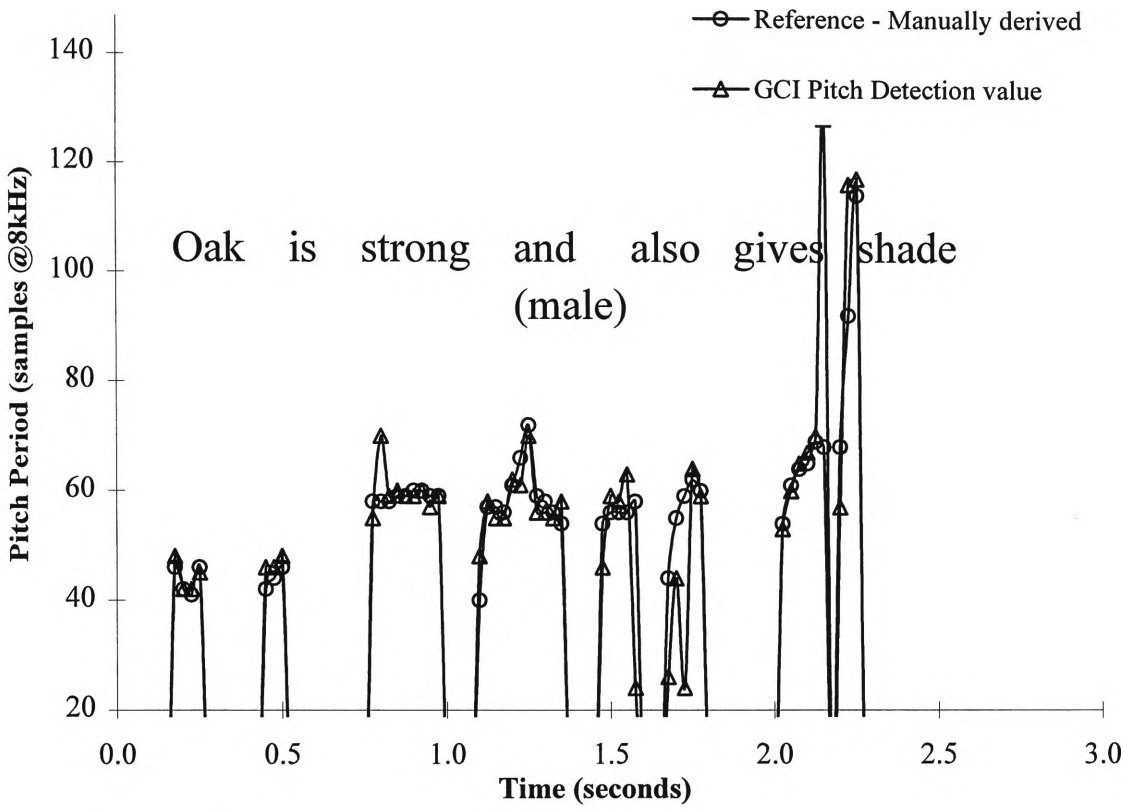

Figure 2.15a - GCI Pitch Detector resultant pitch profile.

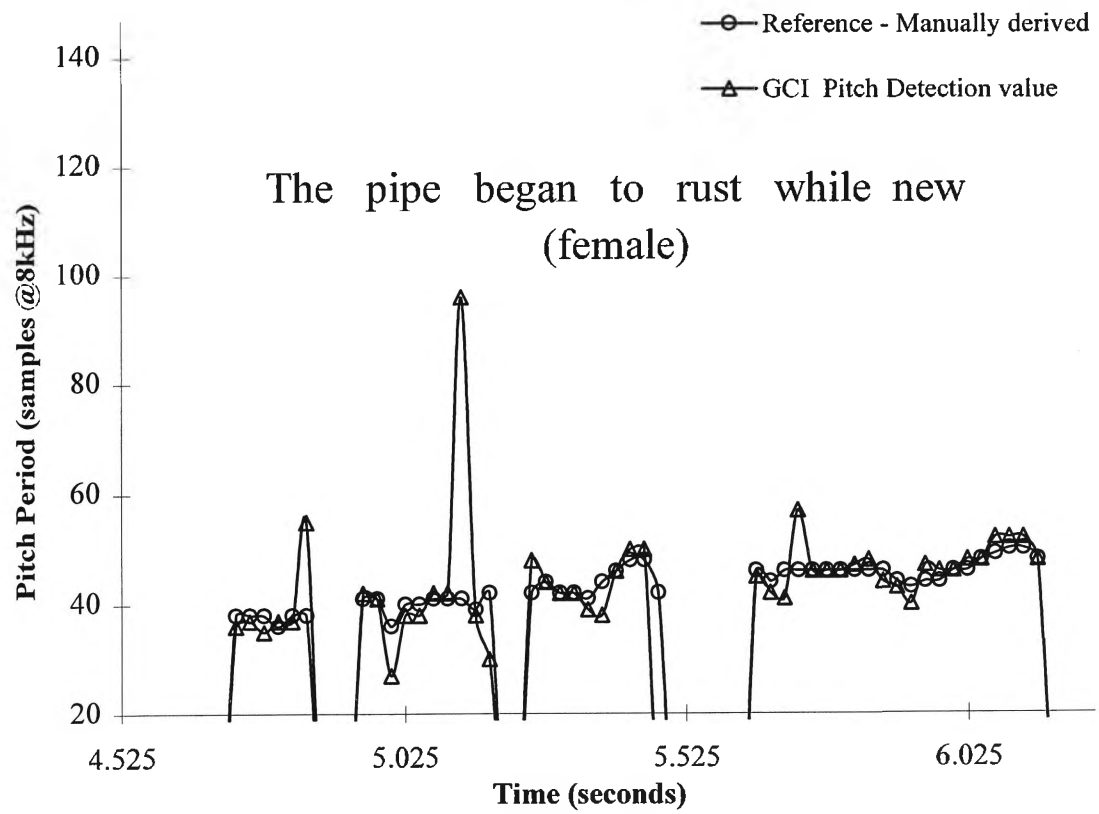

Figure 2.15b - GCI Pitch Detector resultant pitch profile.

2.8 Chapter Summary

The analysis of the speech signal, whereby the speech signal is partitioned into Short-Term and Long-Term components has been presented. Emphasising that Pitch Detection is vital for low bit-rate (2400bps) speech coders. This chapter presented a review of Pitch Detection, commencing with an early real-time Pitch Detector incorporating non-linear clipping operations performed on the speech signal to enhance the Fundamental component. Speech signal characteristics that diminish the detection of the pitch period were highlighted and the approaches taken by alternative methods such as the AMDF were discussed. The Auto Correlation and Cross Correlation based Pitch Detectors remain viable methods in pitch detection. Used alone or in conjunction with preliminary speech processing, these techniques remain reliable, providing for robust techniques in determining the period.

The Simplified Inverse Filter Tracking Algorithm (SIFT) and the Glottal Closure Interval Detection technique were both implemented, with results presented providing a basis for evaluation against other algorithms presented in this thesis. The reliance of the SIFT algorithm on the presence of the Fundamental component, and the inability to discriminate between the Fundamental and it's harmonics are, however, a disadvantage for this algorithm. Both the SIFT and GCI Pitch Detectors are frame-based and, hence, it is difficult to further reduce the algorithmic delay incurred, as they require the entire next frame for processing.

An all-pole transfer function is assumed when modelling the speech vocal tract impulse response. For certain speech characteristics, originating from nasal-coupling or glottal sources, this assumption does not always hold, as they introduce zeros (anti-resonances) into

the transfer function [14]. These speech characteristics are more appropriately modelled by introducing zeros into the vocal tract transfer function [36], however, a more complex (non-minimum phase) model would result. As a consequence of this modelling, the accurate determination of Pitch epochs is required. The Glottal Closure Interval determination technique presented addresses the inability of the all-pole model to accommodate for such speech segments [33]. The Hilbert Transformation is used in the GCI detection method to enhance the Pitch epochs.

CHAPTER 3

PROTOTYPE WAVEFORM

PITCH DETECTION

3.1 Prototype Waveform Pitch Detector

3.1.1 PWPD Introduction

Pitch Detection is an essential component of Prototype Waveform (PW) based coding algorithms [3][4]. A Prototype Waveform Pitch Detection (PWPD) algorithm is presented, which builds on the 'Composite' Auto Correlation Function technique [10],[39]. The algorithm makes inherent use of the PW paradigm and, in this context, a reduction in the look-ahead delay is achieved. The algorithm explicitly considers 'prototype' waveforms within the constituent autocorrelation computations. By detecting the pitch at frequent (5ms) intervals, the algorithm produces a robust and continuous pitch track. The method identifies local 'pitch' by examining prototypes centred on the detection point. By considering all permissible pitch-periods and computing a 'local' pitch at 5ms intervals, the technique offers both accurate pitch-tracking and the capacity to handle transitional behaviour. A reduced-delay algorithm is achieved, which overcomes many of the difficulties encountered when using autocorrelation-based techniques in Prototype Waveform and Waveform Interpolation coders. The following sections of this chapter review the Short-Term 'Composite' Auto Correlation method. Then the improved version of the algorithm developed in this work (termed the PWPD) is presented. Finally, an analysis of the PWPD performance is described and discussed.

Prototype Waveform Pitch Detector (PWPD)

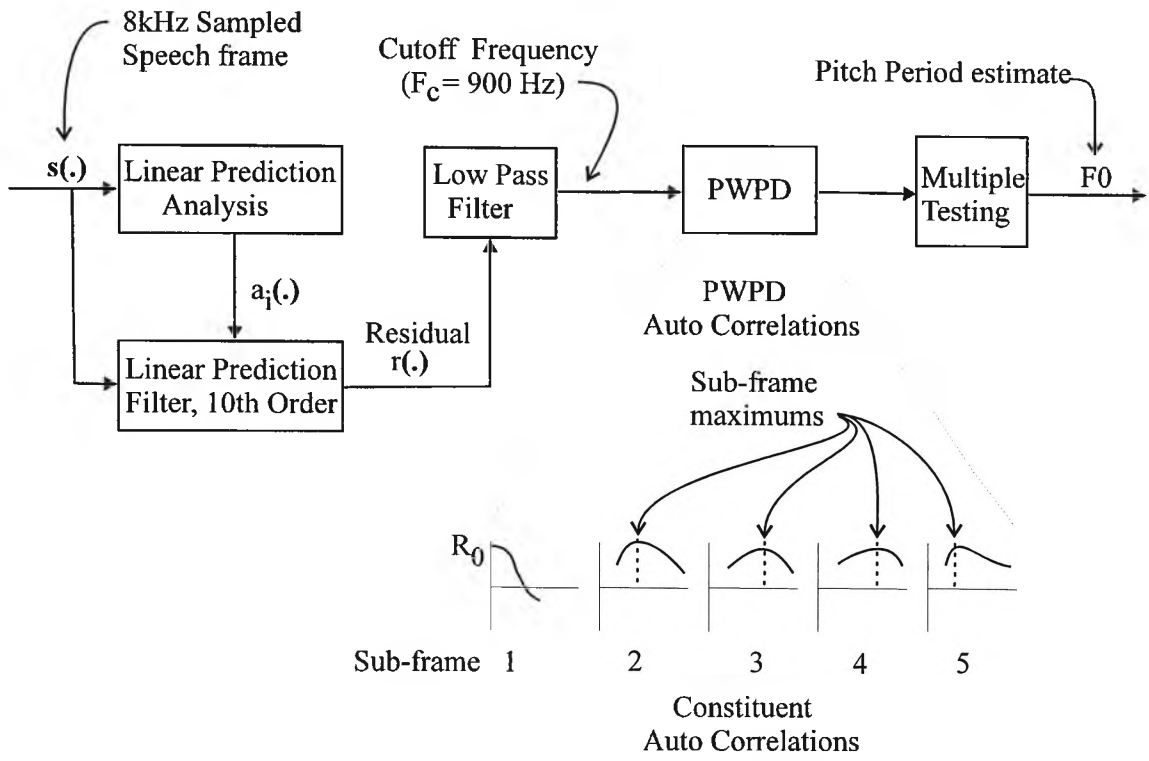

Figure 3.1 - Prototype Waveform Pitch Detector major sub-systems

A block diagram of the Prototype Waveform Pitch Detector is illustrated in Figure 3.1. The proposed method is based on a variation of the pitch detection algorithm as presented in [39]. The constituent Short-Term Auto Correlation Function (ACF) calculation removes unwanted and redundant terms (which exceed the current delay) in the calculation of the Composite ACF, as their inclusion can degrade the ACF estimate [24]. This degradation is caused primarily by the pitch variation across a solitary voiced frame, which can vary a nominal 20% for a "regular" pitch length [7]. The variation across contiguous voiced speech (residual) frames is illustrated in Figure 3.2. By tracking the delay, which yields maximum autocorrelation, the minimum window size can be maintained without compromising the ACF performance. Robustness is therefore provided by the PWPD in that rapid detection of pitch variation is achieved. This is a necessary requirement if prototypes are to be extracted to permit an accurate representation of the pitch cycle waveform.

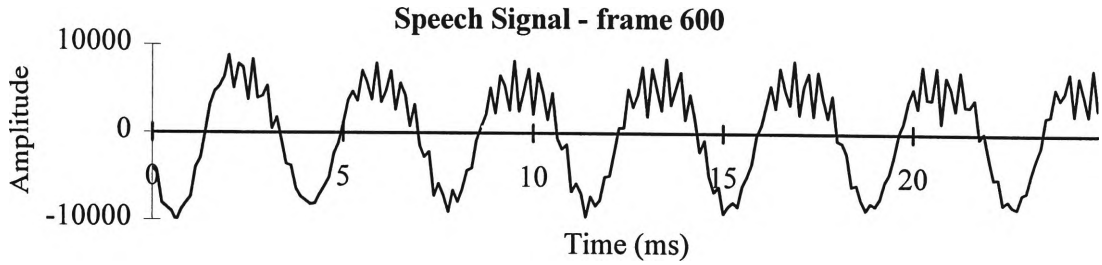

Figure 3.2a - Speech Signal frame 600

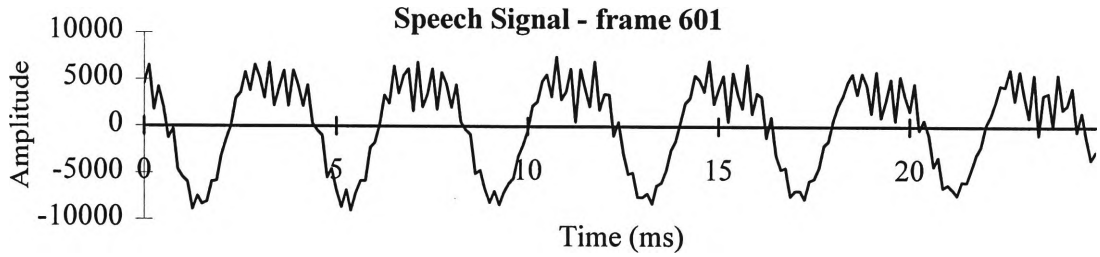

Figure 3.2b - Speech signal frame 601

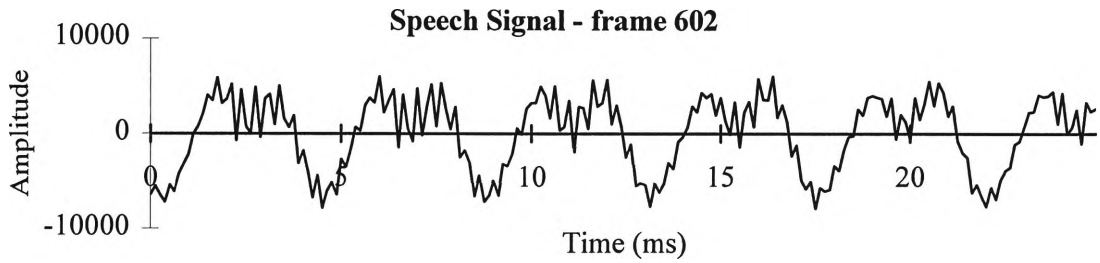

Figure 3.2c - Speech signal frame 602

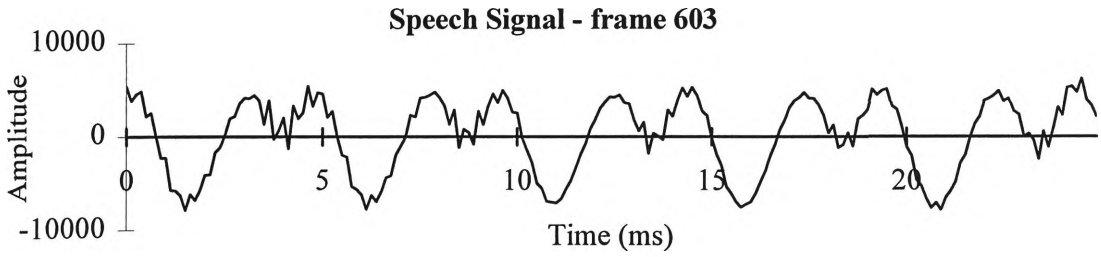

Figure 3.2d - Speech signal frame 603

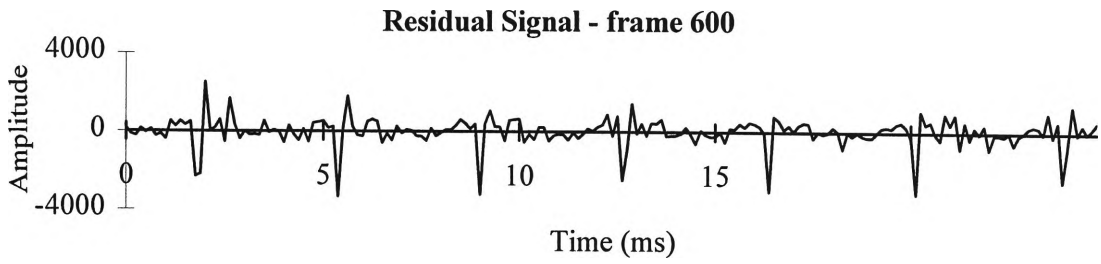

Figure 3.2e - Residual signal frame 600

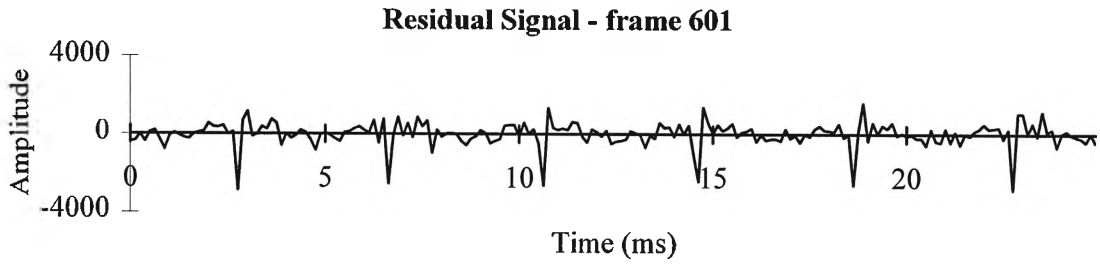

Figure 3.2f - Residual signal frame 601

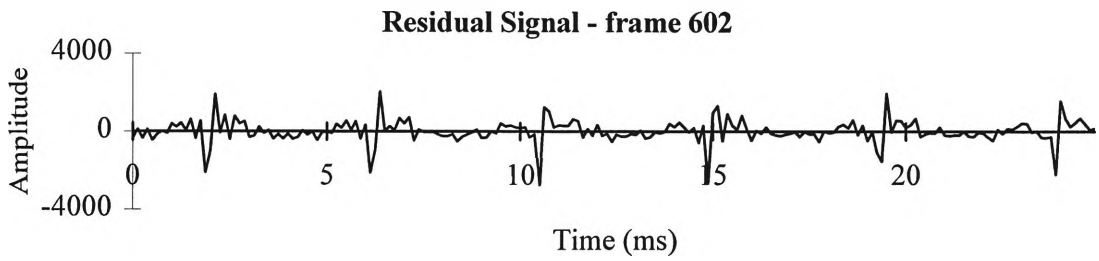

Figure 3.2g - Residual signal frame 602

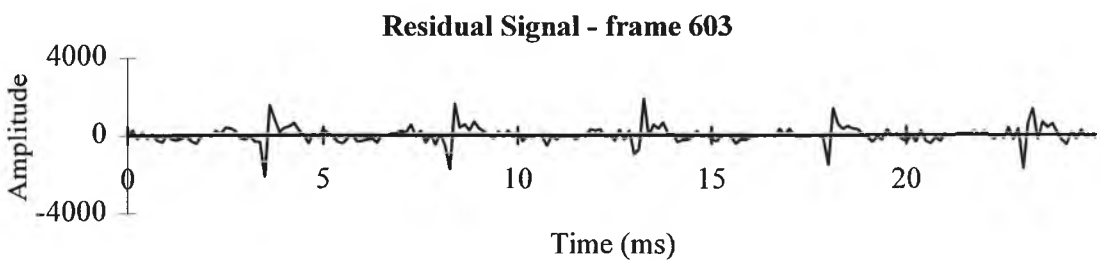

Figure 3.2h - Residual signal frame 603

3.2 Prototype Waveforms

3.2.1 Prototype Waveform Representation

Prototype Waveform (PW) / Waveform Interpolation (WI) encoding of speech has emerged as a viable method of attaining quality coded speech at low bit rates [3],[4],[5],[40],[41]. The base paradigm of PW techniques is that voiced speech, and its residual, exhibits periodic behaviour. This can be efficiently modelled by continuous interpolation of decimated, appropriately extracted, 'pitch-period' sections of the original waveform (the 'prototypes'). For this to be possible, it is essential that the pitch period chosen for extraction is accurate. The results of inaccurate pitch period estimation are false periodicity and perceptually disturbing pitch-behaviour in the reconstructed speech. Pitch detectors based on simple autocorrelation methods may suffer from difficulties which, if ignored, can lead to significant distortions in PW coded speech [7][13][15][24].

Prototype Waveform (PW) interpolation is based on the theory that a band-limited, periodic signal can be represented by its Fourier Series (FS) coefficients [4],[5],[6],[32]. The resulting complex FS coefficients can then be represented as functions of time. The Characteristic Waveform (CW) can be modelled [5] as:

$$u(t,\phi) = \sum_{n} C_n(t).\exp(jn\phi) \quad \text{..... (3.1)}$$

where $C_n(t)$ are the Fourier Series coefficients, and ϕ is the phase.

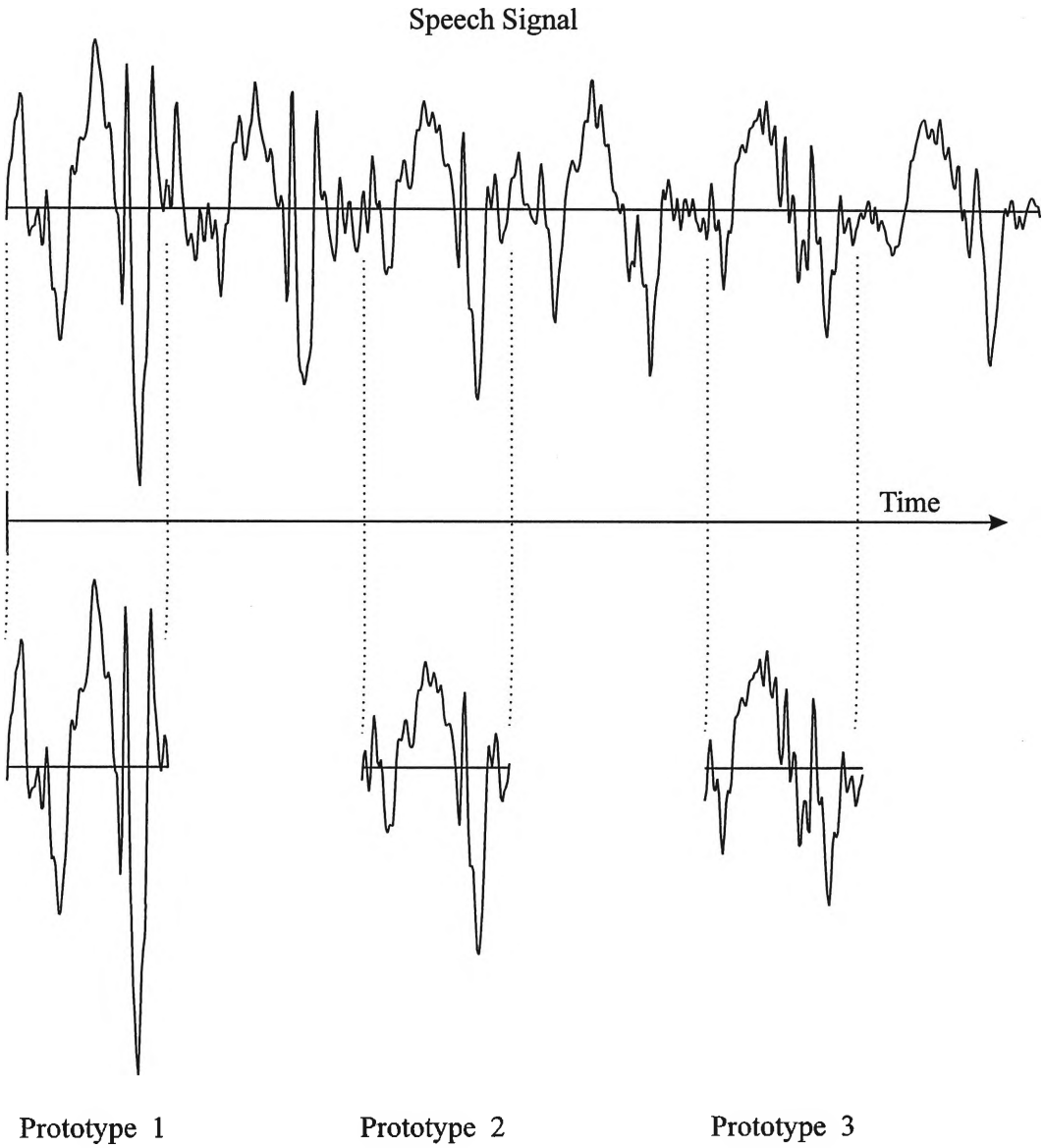

Figure 3.3 - Sample Prototype Waveforms extracted from the speech Signal.

The Characteristic Waveform (CW) can be extracted directly from the speech signal as illustrated in Figure 3.3, however, extraction is simple when performed in the residual domain. The Fourier Series (FS) representation of instantaneous excitation waveforms permits linear interpolation of the pitch period, whereby the linearity may be in either time or

phase. The one-dimensional Linear Prediction (residual) excitation sequence is one instance of the CW at a particular value of the phase(ϕ) and time (t) given by:

$$e(t) = u(t, \phi(t)), \quad where\ \phi(t) = \int_{t_o}^{t} \frac{2.\pi}{p(t')}.dt' \quad \text{..... (3.2)}$$

where $p(t')$ is the pitch period as a function of time.

A FS representation of the CW (the 'prototype') in [4] was transmitted every 20-30ms. At the receiver interpolation was performed in order to return the reconstructed signal back to the original sampling rate of (typically) 8kHz. The low extraction rate of prototypes allows for a significant bit-rate reduction while maintaining a high level of periodicity in the reconstructed speech [5]. An inaccurate pitch period, when interpolated, may result in tonal artifacts distortion ('buzziness'). This increase in periodicity is attributed to the interpolated pitch period not representing the true pitch contour.

3.2.2 Prototype Waveform Interpolation

An initial (1993) implementation of the Prototype Waveform Interpolation (PWI) coder [4] was integrated with a Code Excited Linear Prediction (CELP) coder to account for unvoiced speech segments. The hybrid coder [4] (3500bps PWI, and 4100bps CELP) achieved high quality voiced speech comparable to that of Global Systems Mobile (GSM) and North American full-rate cellular (IS54) standards, at the lower bit-rate. It was concluded in [4] that distortions were due to the transition between the voiced (PWI) and the unvoiced (CELP) segments, with the PWI producing too much periodicity.

In the initial prototype-encoding process, prototypes are required to be extracted from the residual at, ideally, the correct pitch period. The basic idea of the Prototype Waveform (PW) coder is to extract a representation of the pitch cycle [4]. For an initial estimate of the pitch period, it was suggested in [4] that a selection from a variety of algorithms [7] be used. Two methods proposed for the determinatien of pitch period contour were: (1) the maximum Short-Term Linear Prediction (LP) gain criteria. The prediction gain will be relatively large at discontinuites in the speech signal, such as instances of glottal openings (pitch pulses). The sample-to-sample correlation in the speech signal at such instances will be low. The separation between maximum LP gains is a direct measure of the pitch period (closed glottal interval); or (2) the extraction of pitch pulse markers from the up-sampled (U↑10) LP residual [4]. The pitch pulse markers located the concentration of energy. When extracting the residual prototypes, a description of the pitch cycle waveform near the pitch pulse where most of the energy resides was crucial for PWI. The exact location of the boundary was not as important, and the (+/-) 1/2 pitch period on either side of the pitch estimate was sufficient. This latter method was regarded as suitable for its computation reduction, however, a deficiency in this method was the reliance on pitch pulse markers.

3.2.3 Multi-Prototype Waveform Interpolation

Increasing the prototype extraction rate has led to a higher reconstructed signal quality. The increased extraction rate can accommodate a less periodic signal (ie unvoiced speech). To overcome the corresponding increase in bit-rate, an alternative decomposition of the Characteristic Waveform (CW) was performed in [5]. The residual prototype waveforms were

decomposed by low-pass filtering (cut-off frequency 20Hz) and partitioned into two constituent components: (1) Slowly Evolving Waveform (SEW); and (2) a Rapidly Evolving Waveform (REW) [5]. This is accomplished by filtering the time sequence, $C_n(t_i)$, $C_n(t_{i+1})$, $C_n(t_{i+2})$, $C_n(t_{i+3})$,.... for each Fourier Series (FS) coefficient, $C_n(t)$, into respective high and low frequency bands. Lower bit-rates 2.85Kbps (bits per second) are achieved whilst maintaining high quality speech, by selectively coding the SEW and the REW components to meet a desired quality [5].

A similar coder to that in [5] is the Multi-Prototype Waveform (MPW) coder reported in [3]. The MPW coder uses a similar decomposition mechanism for the prototypes, while using the 'Composite' Auto Correlation Function technique to generate the pitch contour. The MPW technique [3] emphasises the accuracy required in Pitch Detection, and concludes that 'the breakdown of the pitch track in transitional Voiced-Unvoiced speech, or in noise, is currently the main obstacle to improved performance'.

3.3 Short-Term Composite Auto Correlation

The 5.85Kbps CELP Algorithm for Cellular Applications in [39] performed interpolation of the Pitch Predictor parameters. The pitch contour was generated using the Short-Term 'Composite' Auto Correlation technique [31]. The 'Relaxed' CELP methods, 'RCELP1' and 'RCELP2', for the closed-loop and open-loop implementations respectively, are based on waveform-matching (time-warping). They provide a mechanism whereby a linear and deterministic delay-contour can be generated, permitting pitch period interpolation. A significant reduction in overall bit-rate was achieved, which does not occur with the Global-

Systems-Mobile (GSM) Cellular Coding Standard [42]. The GSM system is based on Linear Predictive (LP) Coding, and Regular-Pulse Excitation (RPE) with Long-Term Prediction (LTP). Excluding the RPE portion of the overall bit-rate, the LTP Delays and Gains consume 36 bits, which is equivalent to the LP coefficient requirements. Each 20ms speech frame requires 260 bits in total, producing a 13Kbps speech coder, in which the RPE component consumes a large portion of the overall bit-count. In the LTP phase, each speech frame is sub-divided into four sub-frames, requiring four Gain and Delay values per frame to represent the pitch component. The residual signal is reconstructed from the RPE with corresponding delays and gains using:

$$r(k) = G.e(k-d) \quad \text{.....(3.3)}$$

where $r(k)$ is the residual signal, G is the gain, and $e(k-d)$ is the excitation (RPE) with selection delay d.

In [39], only the interpolation of the pitch period for voiced speech segments was performed. The pitch period contour was used as the delay contour for the LTP adaptive-codebook contribution [39]. The RCLEP techniques modified ('time-shifted') the Linear Prediction Residual signal in order to match the adaptive-codebook contribution to the excitation linear delay contour [39]. The fixed-codebook contributions are based on this modified residual signal (the source of the reference entries). Two known problems associated with time-shifting the residual are: (1) the repetition or elimination of residual sections near the shifted boundary which contain a pitch pulse, leading to a corruption of pitch interval and degraded speech quality and; (2) the length of the shifted residual segment is greater than the pitch interval, which then contains two pitch pulses, resulting in a non-optimal match [39]. The

'Composite' Auto Correlation technique [39] presented an alternative method which provided an open loop pitch estimate.

In the 'Composite' Auto Correlation technique each frame was sub-divided into, typically, five equal intervals. For each sub-frame *'i'*, an autocorrelation function, $R_i(d)$, is computed. The composite autocorrelation $R(d)$ is then constructed from the constituent $R_i(d)$ computations. The maximum correlations attained for each delay, *'d'*, calculated across all sub-frames were accumulated to produce the 'Composite' Auto Correlation.

The constituent Auto Correlation Function is given by:

$$R_i(d) = \sum_{n=(i-1)*N_f/N}^{i*N_f/N} x(n).x(n-d) \quad \text{.....(3.4)}$$

where $R_i(d)$ is calculated for all samples *'n'* in the sub interval *'i'* and, N_f is the number of samples per frame. The Composite Auto Correlation Function is then:

$$R(d) = \sum_{i=1}^{N} \max_{l=d-f(k-i)}^{l=d+f(k-i)} \{R_i(d)\} \quad \text{.....(3.5)}$$

where i is the sub-frame index, and k is the selected sub-frame. The function $f(.)$ allows for the variation in pitch period across sub-frames.

3.4 Prototype Waveform-based Pitch Detection

The 'Composite' Auto Correlation Function technique [39] accounted for pitch variation across the frame by dividing each autocorrelation function into several smaller sub-frame calculations. While the 'Composite' ACF technique improves both the tracking and, hence, accuracy of the autocorrelation method, it has been found that the overall accuracy tends to be degraded by the use of the fixed, short sub-frame autocorrelation functions. There is also the necessary requirement to 'look-ahead' one frame if the 'Composite' ACF technique is to successfully track transitional pitch variations.

The fixed sub-frame length in the 'Composite' ACF method ignores the Prototype Waveform (PW) paradigm that the speech residual can be modelled as a train of pitch-period prototypes. The PWPD algorithm builds upon the 'Composite' ACF technique such that the constituent autocorrelations are limited to considering only the (proposed) pitch-period (τ) samples either side of the detection point. In a PW coder, the detection point would also be the prototype extraction point, therefore the tracking of autocorrelations is performed as tests proceed for all detection points. Each constituent autocorrelation is then:

$$R_m(\tau) = \frac{1}{\tau}\sum_{i=0}^{\tau-1} r(N+i)\,r(N-\tau+i) \qquad (3.6)$$

where *r(.)* is the low-pass filtered Linear Prediction Residual and *'N'* is the current detection point.

The composite function is then constructed from the '*M*' constituent autocorrelations such that:

$$R_{const}(\tau) = R_0(\tau) + \sum_{m=1}^{M-1} \max_{d=-f(\tau)}^{d=+f(\tau)} (R_m(\tau_{m-1}+d)) \quad \text{..... (3.7)}$$

where $f(\tau)$ is a function from which the variation in pitch period, allowed between successive extraction points, is derived.

It is important to note that while the first extraction point is tested solely for the local autocorrelation at a period 'τ', subsequent extraction points in the frame are tested for 'τ' within range of the proposed track. The result of this process is a table of possible pitch tracks with 'likelihood' functions stored against them.

3.5 Prototype Waveform Pitch Detector Implementation

3.5.1 Integral Tracking

The principle of a composite maxima (maximum energy), and the corresponding sub-interval which contains a pitch pulse, is used in the 'Composite' Auto Correlation Function (ACF) method presented in [39]. The 'Composite' ACF window size is set proportional to the location of the estimated pitch pulse to account for the pitch period variation. This assumes a limited variation in the pitch period. This assumption is valid for a large percentage of the time but may not account for pitch variation greater than the nominal (+/-20%).

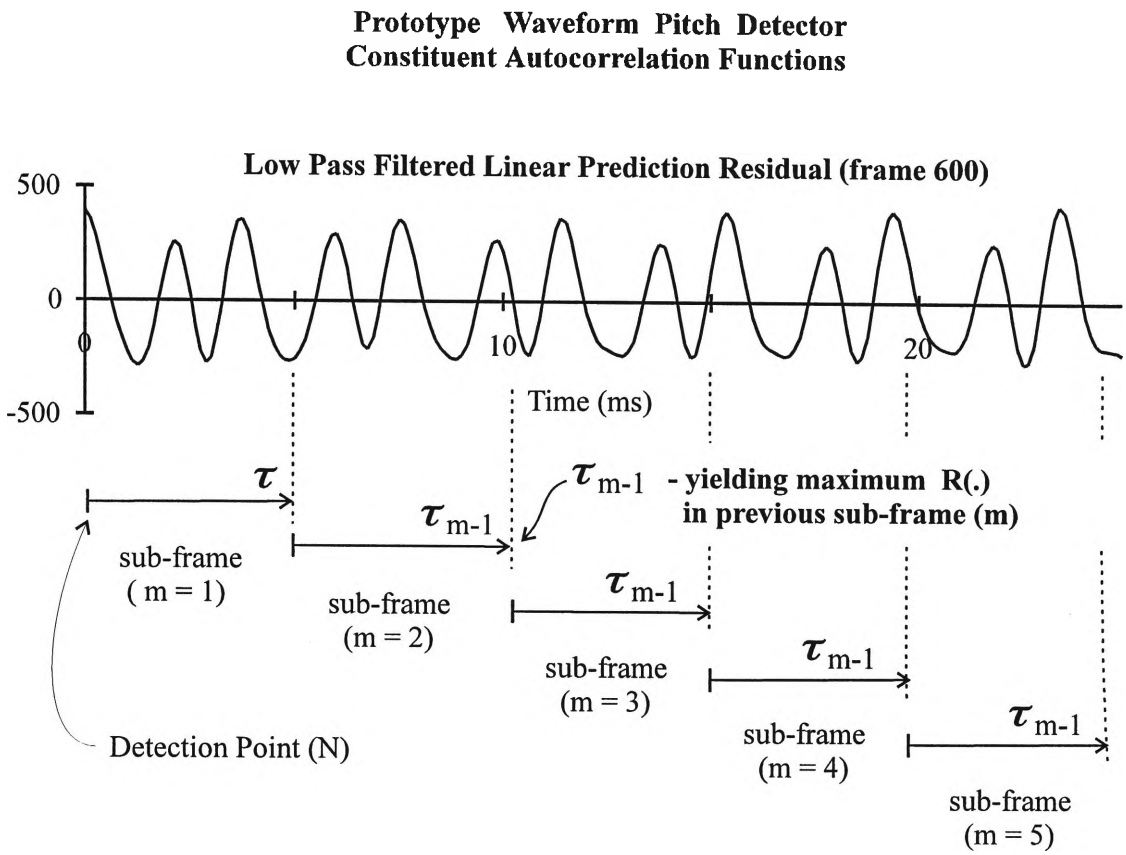

Figure 3.4 - Diagram illustrating the PWPD tracking across constituent autocorrelations.

Tracking the pitch period alleviates the necessity to vary the ACF window size. The inherent tracking of the autocorrelations is achieved, as illustrated in Figure 3.4. The constituent autocorrelations that contribute to the PWPD ACF, at a detection point of 20 samples, for the example frame 600, are illustrated in Figures 3.5a - 3.5e. From these figures, the autocorrelation window is observed to shift in the direction of maximum correlation, tracking the ACF peak. Tracking is continued for the next composite ACF detection point of 21 samples (Figures 3.6a - 3.6e). The ACF in the 5th sub-frame (Figure 3.6e) has recorded a maximum at a 30 sample delay. This example illustrates the rapid convergence of the PWPD. This recorded pitch period of 30 samples is observed to reflect the true pitch period.

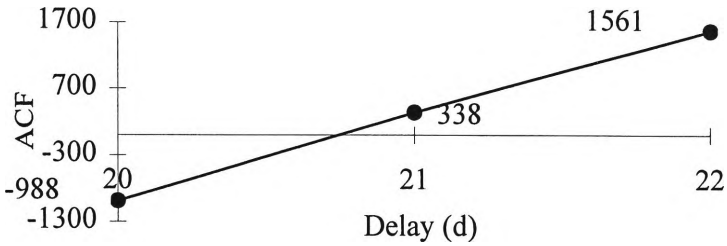

Figure 3.5a - First sub-frame ACF

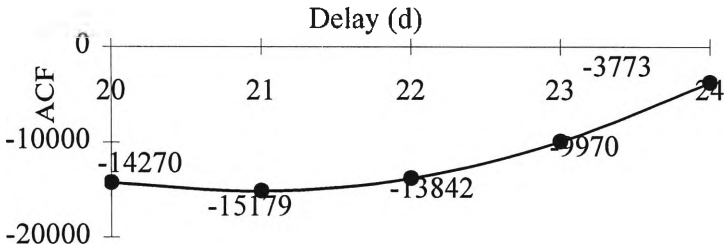

Figure 3.5b - Second sub-frame ACF, window centre is first sub-frame ACF peak

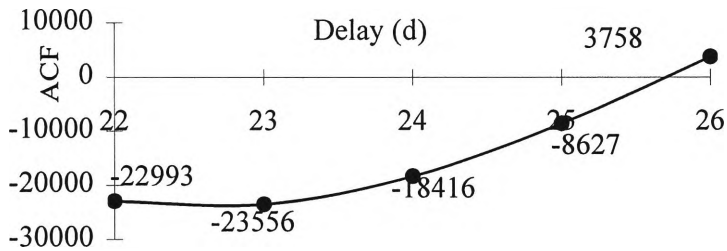

Figure 3.5c - Third sub-frame ACF, window centre is second sub-frame ACF peak

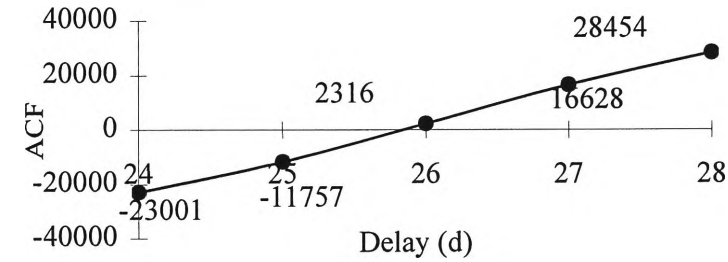

Figure 3.5d - Fourth sub-frame ACF, window centre is third sub-frame ACF peak

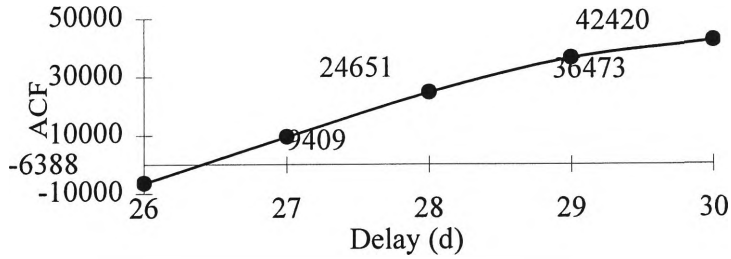

Figure 3.5e - Fifth sub-frame ACF, window centre is fourth sub-frame ACF peak

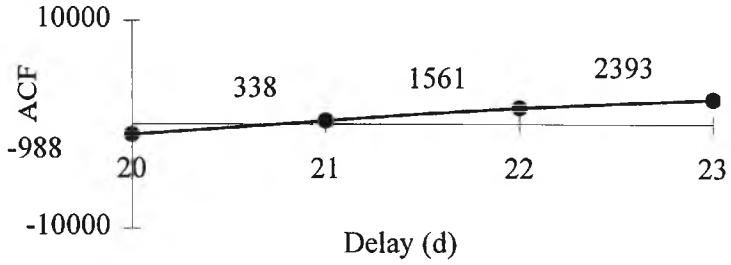

Figure 3.6a - First sub-frame

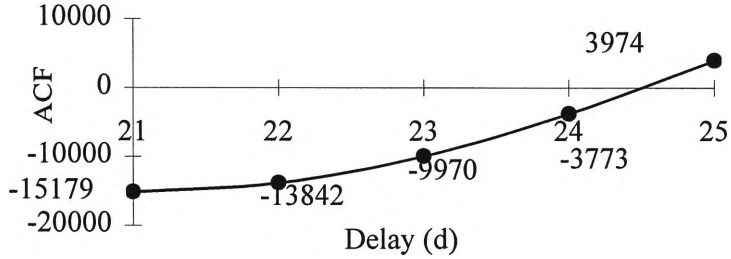

Figure 3.6b - Second sub-frame, window centre is first sub-frame ACF peak

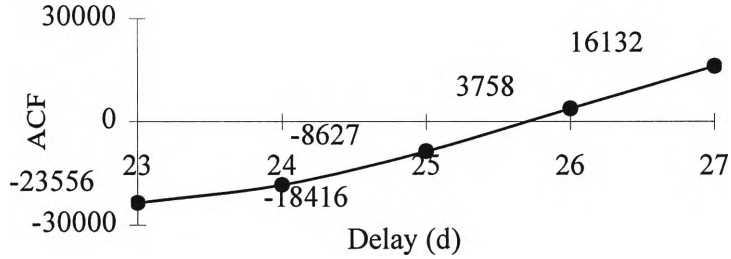

Figure 3.6c - Third sub-frame, window centre is second sub-frame ACF peak

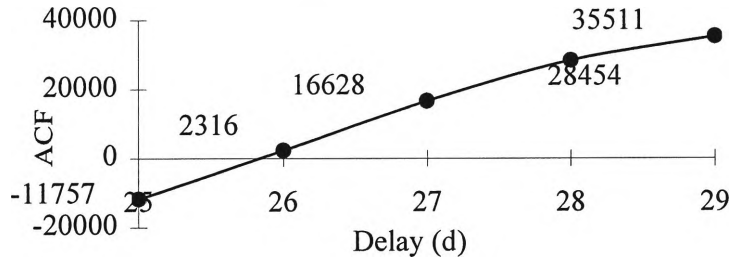

Figure 3.6d - Fourth sub-frame, window centre is third sub-frame ACF peak

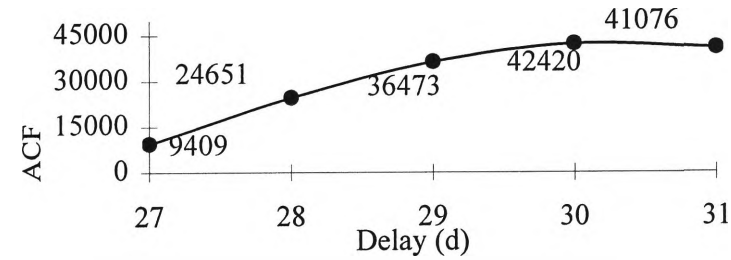

Figure 3.6e - Fifth sub-frame, window centre is fourth sub-frame ACF peak

3.5.2 PWPD Composite Auto Correlation Function

The use of Short-Term ACF windows in preference to Long-Term windows minimises the degradation caused by unwanted signal components (such as the Formant structure that may still be present). The additional tracking capability is achieved as follows: the delay position of the maximum ACF in the current sub-frame is used as the initial delay in subsequent sub-frame ACF computations. The use of the previous delay is represented by the 'τ_{m-1}' term in equation (3.7). A variable window length, which is proportional to the detection point, is not required as in [39], as all possible delays (20 - 147 samples) are tested. The resulting PWPD autocorrelation function, computed by the Prototype Waveform Pitch Detector, for the example (frame 600) is illustrated in Figure 3.7.

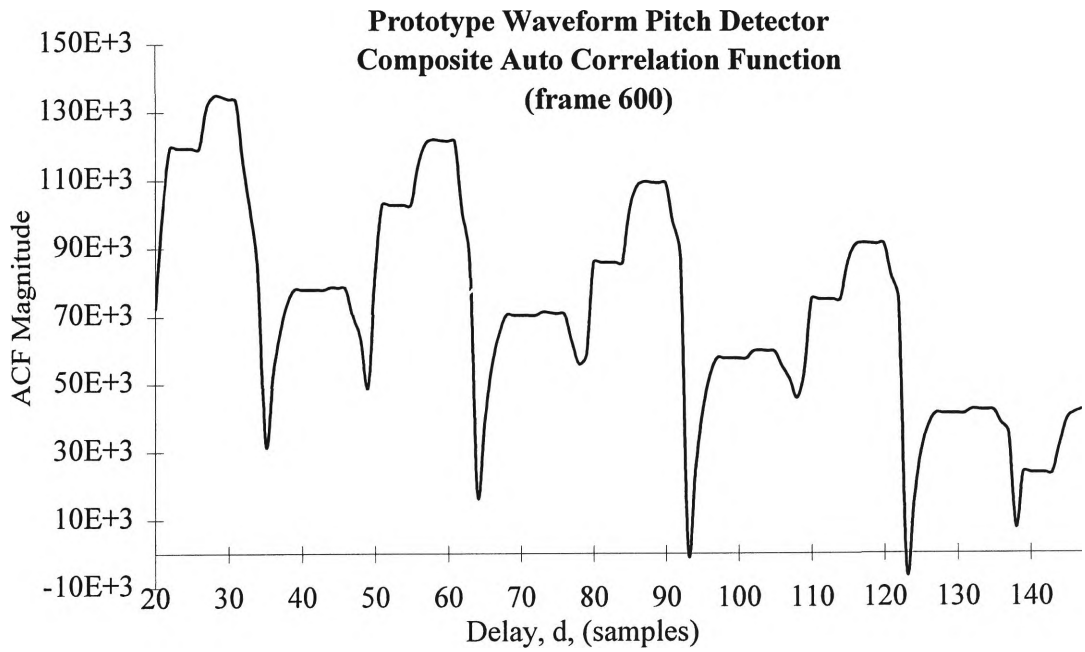

Figure 3.7 - The PWPD Composite Auto Correlation function computed for frame 600. A maximum correlation is recorded at a delay of 30 samples, with harmonics recorded at 60, 90 and 120 samples.

3.6 Prototype Waveform Pitch Detector Results

Some examples of the successful tracking of transitional behaviour by the Prototype Waveform (PW) Pitch Detector are illustrated in Figures 3.8a-b. A complete set of figures are contained in Appendix C. The reference pitch profile was generated manually from a data base comprising 20 female and male 'British' English language speakers [35]. While the algorithm has a high overall success rate in tracking both steady and transitional behaviour accurately, some minor problems still exist with onsets of voiced speech for certain male speakers. This is believed to be caused by the fact that long pitch periods make the operation of the PW pitch detection algorithm similar to a 'Composite' Auto Correlation Function operation. Table 3.1 presents a breakdown of the algorithm's performance with respect to Fine and Gross pitch errors.

Table 3.1

Results for the Prototype Waveform Pitch Detector applied to a 2024 frame speech data base of which 1079 are considered voiced.

Error Classification	Count	Average (samples)	Std. Dev. (samples)
Voiced frames with no Errors	241	0	N/A
Voiced frames with Fine Errors [≤ 8 samples (≤ 1ms)]	763	2	1
Voiced frames with Gross Errors [> 8 samples (> 1ms)]	75	42	34
Total voiced frames	1079	4	16

3.7 Conclusion

This new technique offers a number of advantages: by limiting the correlation computation in the constituent functions to a single 'prototype length', it limits the possible pitch period alteration (and hence distortion) included in the computation. Further, this maximises the result of the normalised correlation, since multiple pitch periods (which vary in shape) are not included in the constituent calculations. Checks for pitch doubling and halving are minimised to allow tracking of natural pitch events. Tracking of the pitch is also inherent in the computation of the composite function, allowing for smoother pitch tracks. The technique detects the pitch based on the correlation of 'τ' length blocks on either side of the detection point. The look-ahead delay required will be equal to the maximum 'τ' minus the time between detection points (14ms for a 5ms detection point separation, with a maximum 'τ' of 150 samples). This is approximately a 30% improvement over the whole frame look-ahead delay, which is required if the autocorrelation method is to successfully track transitional behaviour. The ability of the algorithm to rapidly converge to the required pitch is highly desirable, and lends itself to the extraction of prototype waveforms for coding.

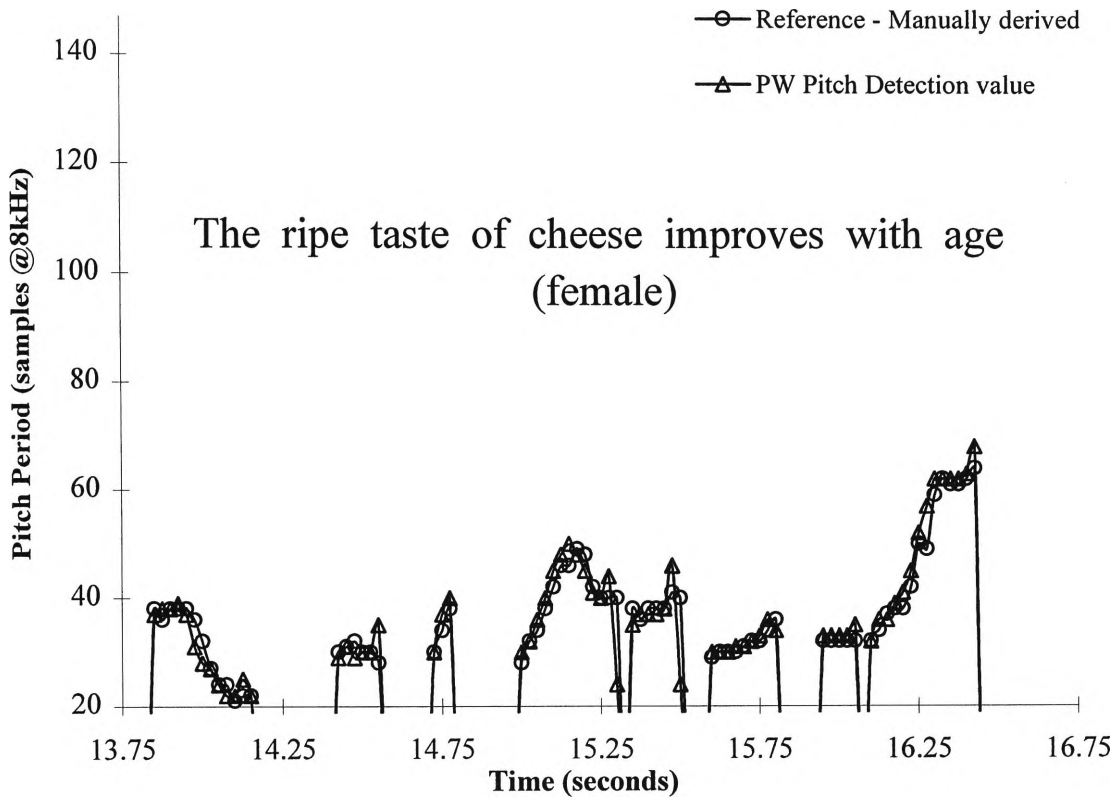

Figure 3.8a - Prototype Waveform Pitch Detector generated Pitch Profile

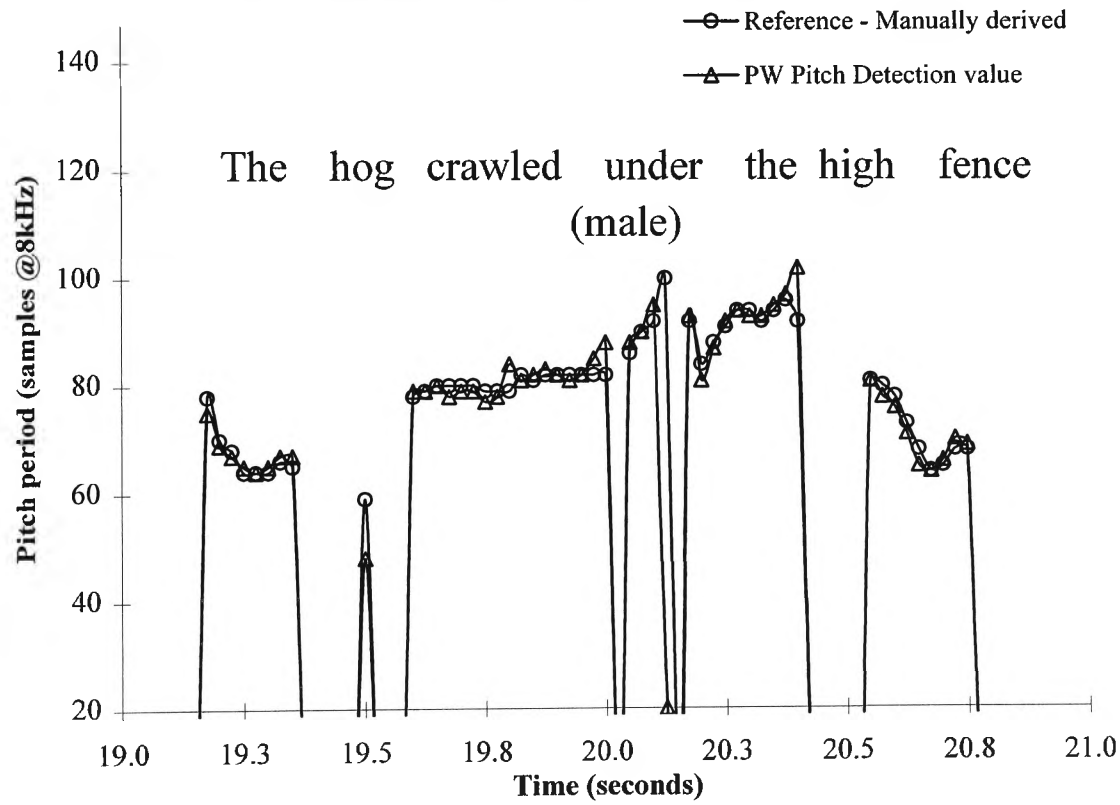

Figure 3.8b - Prototype Waveform Pitch Detector generated Pitch Profile

CHAPTER 4

DYNAMIC PROGRAMMING / VITERBI PITCH DETECTION

4.1 Introduction

This chapter presents an algorithm for the detection of the Fundamental Pitch Period (F_0) at low delay, utilising Dynamic Programming (DP) and the techniques of the Viterbi Algorithm [43], [44], [45]. The algorithm results in robust, accurate, and low-delay Pitch Detection, which is vital for the attainment of high-quality, low bit-rate speech coders. These speech coders include Prototype Waveform Interpolation (PWI) [5] based methods. Current pitch detection methods based on autocorrelation incur a significant look-ahead delay and, thus, do not offer an optimal solution for use in such speech coders [7]. The algorithm described uses improved, non-linear Pitch Detection, with a substantial extension of a Viterbi-type tracking algorithm, to maintain a smooth pitch track [46].

The constraints placed on the existing Dynamic Programming (DP) algorithms were found to be inadequate in determining optimal pitch tracks. In particular, a purely DP-based algorithm is vulnerable to pitch doubling and tripling effects (large excursions from pitch track), resulting in the requirement for restrictions to be placed on track deviations. The ambiguity in selecting the optimal track also leads to the incursion of a significant processing delay prior to the final pitch decision. This DP delay would result in DP-based Pitch Detection being classified as inappropriate for low delay speech coders. To reduce this delay, the Pitch Detection algorithm presented in this chapter incorporates a finite number of Partial Error Criteria (PEC) [46] minima, as a result of the DP process. These minima are transformed into Viterbi states for each speech sample and inserted into a Viterbi-type trellis. Over a speech segment this generates a selection of weighted paths from which an optimal path is selected. Figure 4.1 illustrates the major sub-systems that constitute the DP/V Pitch Detector.

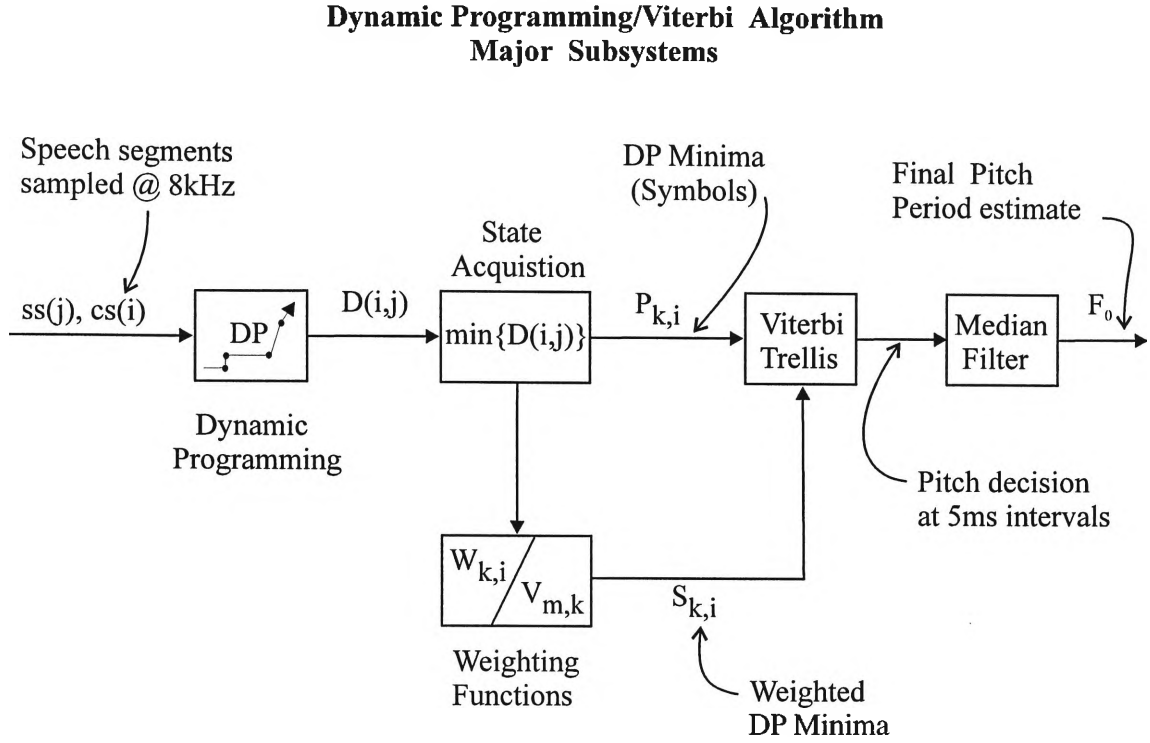

Figure 4.1 - Block diagram of the DP/V Pitch Detector

The speech signal, sampled at 8kHz, is divided into 40 sample (5ms) segments. For each sample in the segment, *cs(i),* a vector of correlations with surrounding samples, within realistic pitch period limits (20-147 samples or 400-54Hz), is computed. Based on the correlation vector, the Dynamic Programming (DP) algorithm is used to form a 'likelihood vector', which incorporates information from the surrounding samples in the two-dimensional DP 'time-warped' plane, *D(i,j).* This 'likelihood vector' is then used as an input to a Viterbi-type process. From the vector, a finite set of minima are extracted (State Acquisition), weighted, and then input into a Viterbi-type trellis. This leads to a set of 'weighted' possible pitch tracks, expressed as paths in the trellis. The pitch decision is based on the accumulated weights of prospective paths in the trellis. The advantage of the Viterbi-type trellis is that redundant, and potentially misleading, information from the DP algorithm can be removed by using path likelihood. This is a result of the algorithm considering sample-by-sample pitch behaviour, while utilising the concepts of natural variation.

4.2 Dynamic Programming

4.2.1 Principle of Optimality

Dynamic Programming (DP) and the 'Principle of Optimality', which is attributed to Bellman [43], form the bases for the algorithms presented in [46][47][48]. These algorithms applied DP principles to Spoken Word Recognition and Pitch Detection. The 'Principle of Optimality' can be stated as follows: "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision" [46].

Using the 'Principle of Optimality', Dynamic Programming (DP) performs the minimisation of the function, $D(i,j)$, representing a Partial Error Criteria (PEC) [46]. The PEC constitutes a two dimensional time-warped plane created from two speech segments that can be matched for similarity [46]. This time-warped plane provides the mapping of the signal segment $cs(i)$, (reference signal) to that of a similar signal segment $ss(j)$. In speech recognition, the latter speech segment is, typically, the unknown speech segment, whereas in pitch detection both signal segments constitute the same speaker. The speech segments are adjacent in time, separated by the minimum pitch period interval, hence the indices 'i' and 'j' traverse the same data signal, as illustrated in Figure 4.2.

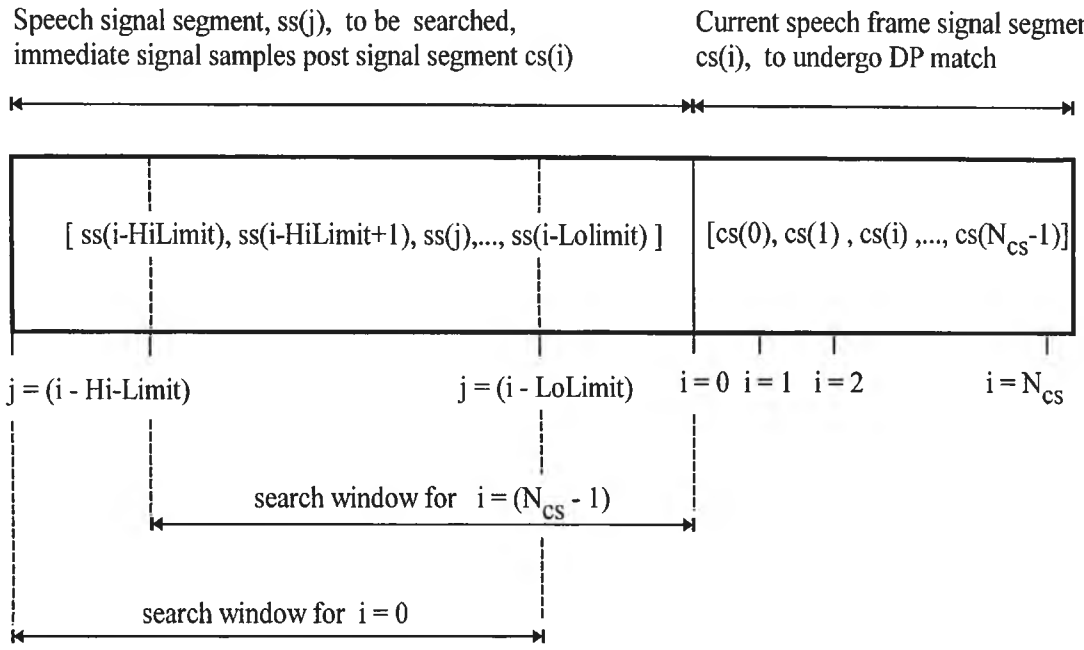

Figure 4.2 - Adjacent speech signal segments to be matched using Dynamic Programming. The speech signals are separated by the minimum pitch period (LoLimit).

The DP objective is to optimally map the reference signal, *cs(i),* to the latter similar signal *ss(j),* based on the minimisation of an accumulated residual error. This is achieved by the Partial Error Criteria (PEC), which is defined as the minimum accumulated sum of errors from an unknown point, $i = 0$, (j = *unknown*), to the point *(i,j)* in the time-warped plane. The non-linear calculation of *D(i,j),* performed over the required interval, is defined as:

$$D(i,j) = \min_{\{w\}} \left\{ \sum_{k=1}^{\lambda} d(w(k) : w(\lambda) = (i,j) \right\} \quad \text{.....(4.1)}$$

where *'W'* is the two-dimensional time-warped plane, $d(w(\lambda))$ is the selected distance function, and (i,j) is a point within the plane. The function $w(\lambda)$ is, therefore, the required mapping function.

In contrast, a linear strategy reduces to:

$$D(k) = \min_{\{all\,k\}} \left\{ \sum_{k=1}^{\lambda} d(k) \right\} \quad \text{.....(4.2)}$$

where $d(k)$ is a linear distance function, as used in the AMDF [17].

4.2.2 Optimal Solution Derivation

In providing a solution to (4.1), the Dynamic Programming (DP) approach considers the minimisation of an N-stage discrete deterministic Markov process. Laying aside a traditional numeric solution to this problem, DP instead applies numeric techniques employing the Markov Model to arrive at a solution [49],[50],[51],[52].

The N-stage Markov process, $F_N(p,q)$, is subsequently separated into its constituent functions and expressed as:

$$F_N(p,q) = g(p_1, q_1) + g(p_2, q_2) + \ldots + g(p_N, q_N) \quad \text{.....(4.3)}$$

The arguments (p_N, q_N) are the state and decision variables at stage *'N'* respectively.

The latter constituent functions (p_n, q_n) represent the minimisation return, excluding the return from previous (p_{n-1}, q_{n-1}) stages. The resulting expression for the N-stage decision process is therefore:

$$f_N(p_N, q_i) = \min_{\{q_1\}}\left[g(p_1, q_1) + f_{N-1}(p_{N-1}, q_i)\right] \quad where, 1 \le i \le M(i) \text{.....(4.4)}$$

where *M(i)* is the observation space for the *'i'th* stage decision, and given that

$$p_n = f(p_{n-1}, q_{n-1}), \; for \; n = 1..N \text{ .....(4.5)}$$

and its substitution into (4.4), obtaining,

$$f_N(p_N, q_i) = \min\left\{g(p_1, q_1) + f_{N-1}(f(p_N, q_i))\right\} \text{.....(4.6)}$$

then (4.6) provides the solution incorporating the 'Principle of Optimality'.

4.2.3 Reduced Form Dynamic Programming Algorithm

In the context of the two dimensional time-warped plane used in the Dynamic Programming (DP) method, equation (4.1) is expanded into its constituent functions resulting in:

$$D_N(i,j) = d(i_1, j_1) + d(w(i_2, j_2)) + ... + d(w(i_N, j_N)) \text{.....(4.7)}$$

In applying DP to the determination of an optimal match between two speech signal segments, a set of constraints have been imposed [46],[47],[48] on the warping function $w(\lambda)$ to enable a practical solution to be reached. These constraints have resulted in a simplified relationship, or Reduced Form, of the DP algorithm amenable to practical implementation.

The Constraints placed on the warping function are:

(a) Boundary Condition: $w(\lambda)$ must lie between the minimum and maximum pitch period;

(b) Monotonic Condition: the chronological sequence of signal samples should be retained.

(c) Continuous Condition: all signal samples should be utilised.

The conditions (b) and (c) combine to give a continuous constraint applied to all calculated points in the time-warped plane, and the mapping function $w(k)$ of the DP equation reduces to:

$$w(k-1) \in \{(i-1,j),(i-1,j-1),(i,j-1)\} \quad \text{.....(4.8)}$$

Observing (4.8), a significant reduction in the observation space of $w(k)$ is achieved. Further, the substitution of (4.8) into (4.7) yields:

$$D(i,j) = d(i,j) + \min\{D(i-1,j), D(i-1,j-1), D(i,j-1)\} \quad \text{.....(4.9)}$$

The 'min {...}' term in equation (4.9) differentiates this method from standard linear search techniques inherent in methods such as the ACF and AMDF. Surrounding samples are not

considered in linear techniques, as they are in Dynamic Programming (DP). In the DP method, the delay *'d'* between corresponding samples of the two signal segments undergoing the match function need not be constant. This significant difference is illustrated in Figure 4.3.

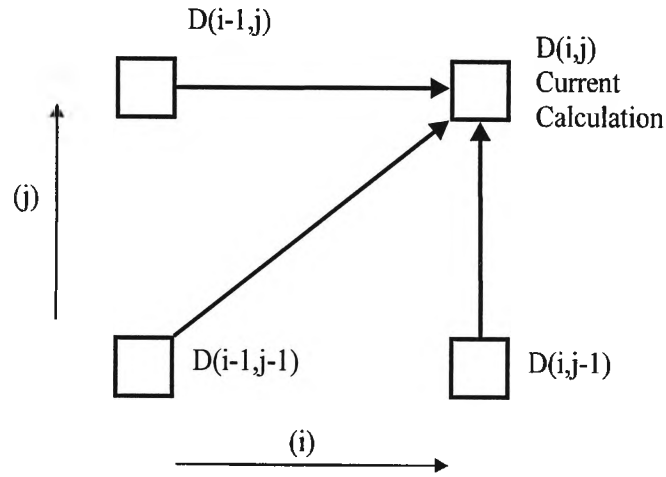

Figure 4.3 - Reduced Form DP algorithm illustrating the signal samples that are used.

The time-warped plane *D(i,j)* values are computed iteratively. For each speech sample *'i'* of the signal segment, *cs(i),* to be matched, *D(i,j)* is calculated for all allowable pitch period values, *'j'*, in the similar signal segment, *ss(j),* to be searched. The calculation takes into consideration all the permissible paths, to arrive at the current *D(i,j)* state. Neighbouring values, including previously calculated values such as *D(i-1,j-1),* are used in determining the current *D(i,j)* value. This is illustrated in (4.9) by the arguments in the 'min' function.

4.2.3.1 Avoiding Unwanted Signal Samples

As a consequence of the constraints placed upon consecutive samples in the time-warped plane, only those immediate samples surrounding the current sample are taken into

consideration. This process clearly illustrates that unwanted spurious signal samples may be removed once it is determined that alternative samples will yield a lower residual error (PEC). The ability to avoid considering unwanted signal samples is a benefit in Pitch Detection. By incorporating the DP algorithm, the optimisation on a sample-by-sample basis occurs (equation 4.9). This will benefit pitch detection when the speech signal is corrupted by noise. This has been demonstrated with the implementation presented in this thesis. The key concept underlying this is that the delay between signal samples does not need to be constant.

4.2.4 A Slope Constraint

The Dynamic Programming (DP) algorithm has been applied with a 'slope constraint' to allow relaxation on deviations that can occur to a path as it is tracked through the time-warped plane [47]. The 'slope constraint' is a gradient range given by the ratio of the horizontal or vertical direction, distance to that of the diagonal direction distance, permitted by tracks traversing the time-warped plane. This is achieved by the introduction of additional *D(i,j)* values into the DP 'min' function, significantly altering the DP algorithm. The addition of *D(i,j)* values from the time-warped plane maintains avoidance of the non-permissible paths, as defined (Section 4.2.3) by the constraints placed on the PEC function. In spoken word recognition systems a significant relaxation, for each speech sample being matched, was permitted and, which extended to (+/-)108ms. For a more detailed reference on the application of the DP principles, developed by Sakoe & Chiba [46] for Connected Speech Recognition and Discrete Utterance Recognition, see Silverman & Morgan [47]. This large window is considered inappropriate in a DP-based Pitch Detection algorithm and thus the minimum window size of 0.25ms (2 samples) is used. In the Dynamic Programming/Viterbi (DP/V) Pitch Detection algorithm presented the slope constraint is maintained by allowing

this gradient to be unity, (the optimal value [47]). This enabled the extended DP algorithm to be used as shown below:

$$D(i,j) = d(i,j) + \min\begin{Bmatrix} D(i-1,j-2) + 2d(i,j-1), \\ D(i-1,j-1) + 2d(i,j) \\ D(i-2,j-1) + 2d(i-1,j) \end{Bmatrix} \quad \text{..... (4.10)}$$

Source states that stem from a diagonal are scaled, by a factor of two, to bias such preferred paths. The Modified DP algorithm is shown in Figure 4.4 with an illustrative example in Figure 4.5

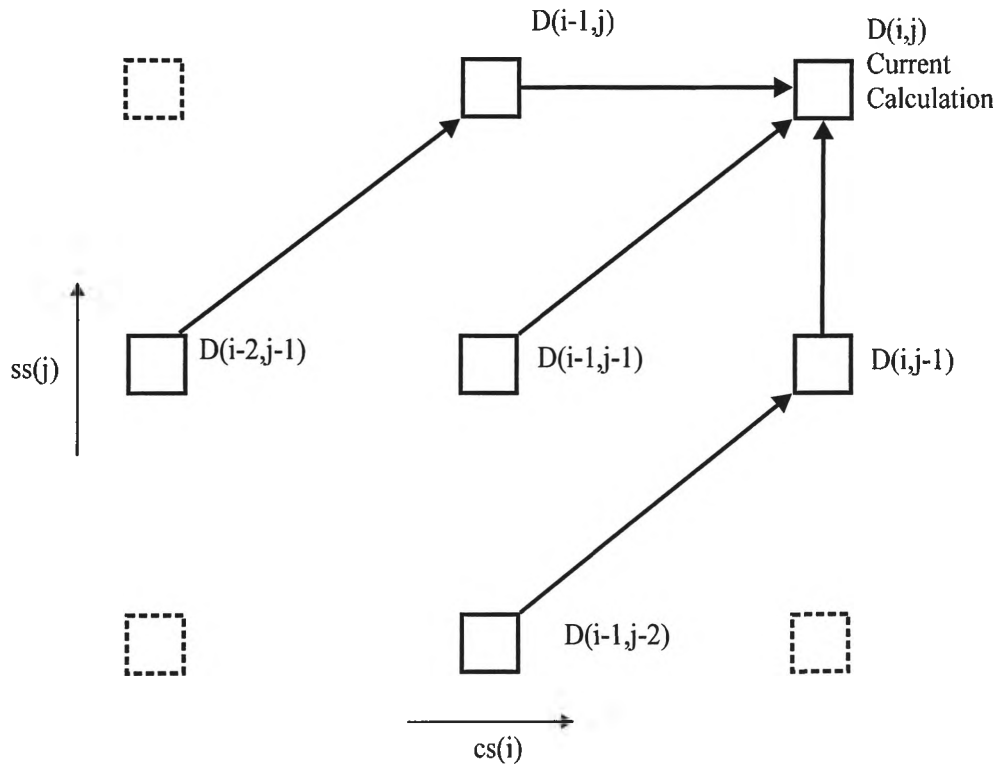

Figure 4.4 - The DP algorithm used in the DP/V Pitch Detection method, illustrating the increased number of previously computed *D(i,j)* values taken into consideration in determining the current *D(i,j)*.

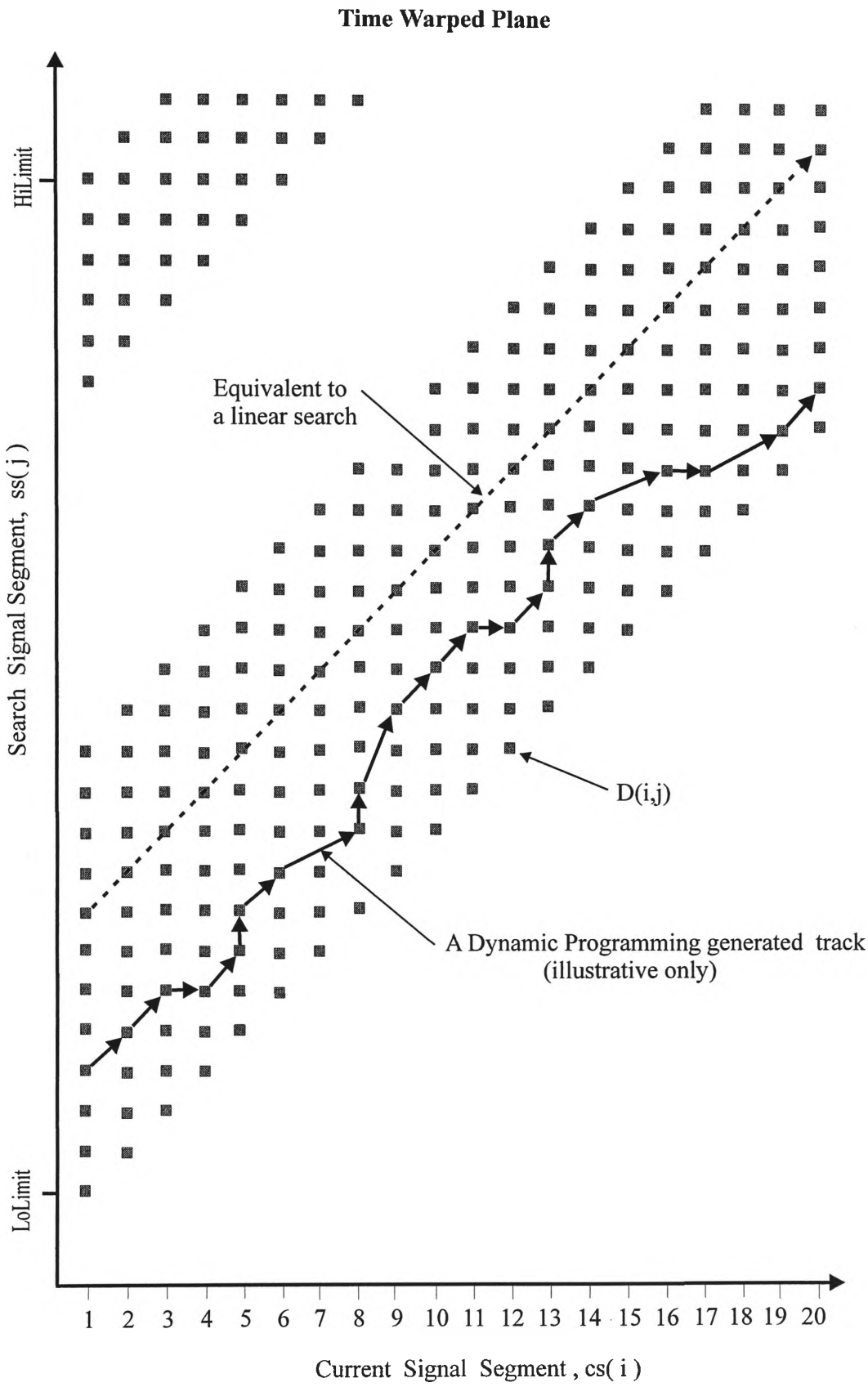

Figure 4.5 - Time-warped plane illustrating the DP generated track.

4.2.5 Dynamic Programming Distance Function

The distance function used in the Dynamic Programming method, up until this point, has simply been denoted by *d(.)*, which assumes a generic distance function. In [7] it is shown that the distance function is not dissimilar to those used in other pitch detection algorithms. The distance functions used by a number of time-domain Pitch Detection techniques may be classified as belonging to a generic class of distance functions [7]. The Average Magnitude Difference Function (AMDF) Pitch Extractor (4.11) is considered, as an example [17].

$$AMDF(d) = \frac{1}{k} \sum_{n=q}^{q+K-1} \| s(n) - s(n+d) \| \quad \text{..... (4.11)}$$

where *s(n)*, denotes the speech signal and *'d'* is the delay.

The Generalised Distance Function is given by:

$$D(d,k) = \left\{ \sum_{n=q}^{q+K-1} \left[\, | s(n) - s(n+d) | \, \right]^k \right\}^{\frac{1}{k}} \quad \text{..... (4.12)}$$

The AMDF method belongs to this generic class of distance functions (k = 1). In the Dynamic Programming (DP) based method, the distance function used also belongs to this class. To further illustrate this generic distance function, setting k = 2 will yield the average squared difference function, or the Mean Square Error criteria, when normalised. The similarity between the AMDF and the DP *D(i,j)*, however, ends at this point. The summation that

occurs in the DP algorithm is performed non-linearly over the required interval, and not contiguously as n = q, (q+1), (q+2),...,(q+K-1) (as in linear summations).

4.3 Optimal Pitch Track Determination

The Dynamic Programming (DP) algorithm, when used alone, has been demonstrated to determine the pitch period of speech in a non-optimal manner [46][48]. One reason for this is the selection of solitary minima and subsequent poor tracking [46] caused by window and slope constraints [47]. In addition, the PEC values *D(i,j)* (calculated from a raw or filtered speech signal) may contain unwanted signal samples that contribute to an incorrect track selection. These unwanted samples may be due to signal corruption as a result of noise. Alternatively, a large deviation in pitch track may be attributed to pitch period doubling or tripling, either as a consequence of a natural speech characteristic, or the presence of a dominant harmonic of the Fundamental or Formant frequency. An alternative solution to maintaining the track of the optimal path is now presented. The method presented is based on retaining a finite number of PEC minima during the DP process. This finite set of minima accounts for the optimal pitch period track, and also those of secondary and alternative pitch candidates.

By observing the DP algorithm output, large deviations in the minima are revealed, resulting in an ambiguity in track selection, confirming the poor tracking capability. Observations of large track deviations reveal that when the second and subsequent PEC minima are recorded, tracks can exist across the resultant array of minima (sample to sample). This is a result of the influence of harmonics of the fundamental period and higher Formant frequencies. These observations are illustrated in Figure 4.6.

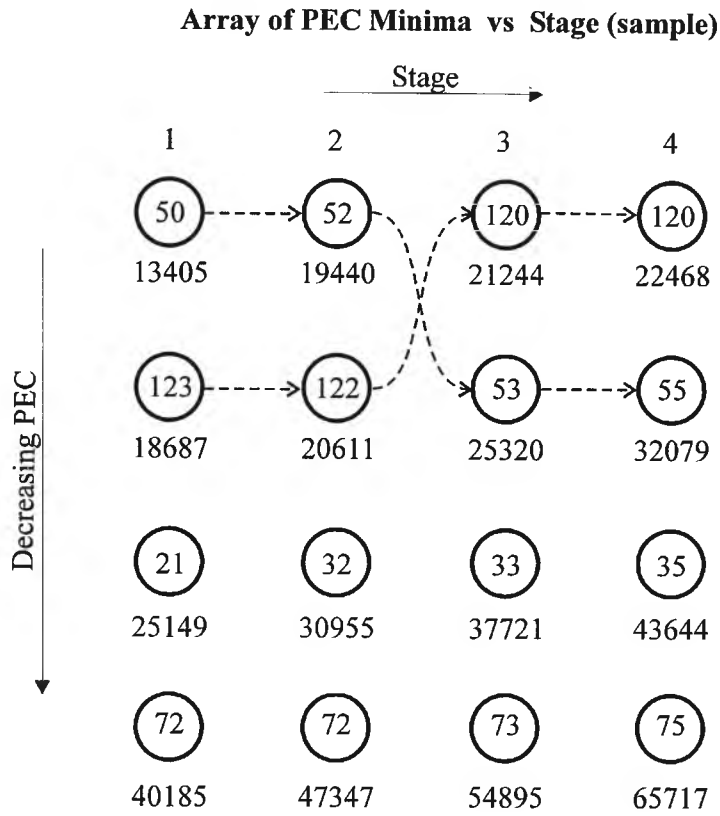

Figure 4.6 - Array of DP derived minima (PEC scores), extracted from frame 49 (first sub-frame). The array illustrates the large deviation in the DP-recorded minimum. The positions at which DP minima occur are also indicated within the array elements, and the trace of paths through this array is observed.

The influence of these large deviations is not so predominant once it is recognised that the remaining minima are valid results. Retaining these minima provides the necessary information to maintain smooth pitch tracks from which an optimal track is selected. In the work undertaken in this thesis, the Viterbi algorithm was utilised to maintain a number of possible tracks within a trellis, accumulating likelihood scores against each track. The most likely paths were retained for the duration of the signal segment under test, and a subset of these, with appropriate adjustments to the scores, were further retained and propagated for use with the next signal segment. The Viterbi algorithm overcomes the current situation whereby

a non-optimal path is selected and tracked. This is achieved by retaining a finite set of DP-derived minima (as states) within a Viterbi-type trellis. To illustrate how the Viterbi trellis is used for the maintenance of candidate pitch tracks, four tracks (survivor paths) connecting the states over the duration of four signal samples are shown in Figure 4.7.

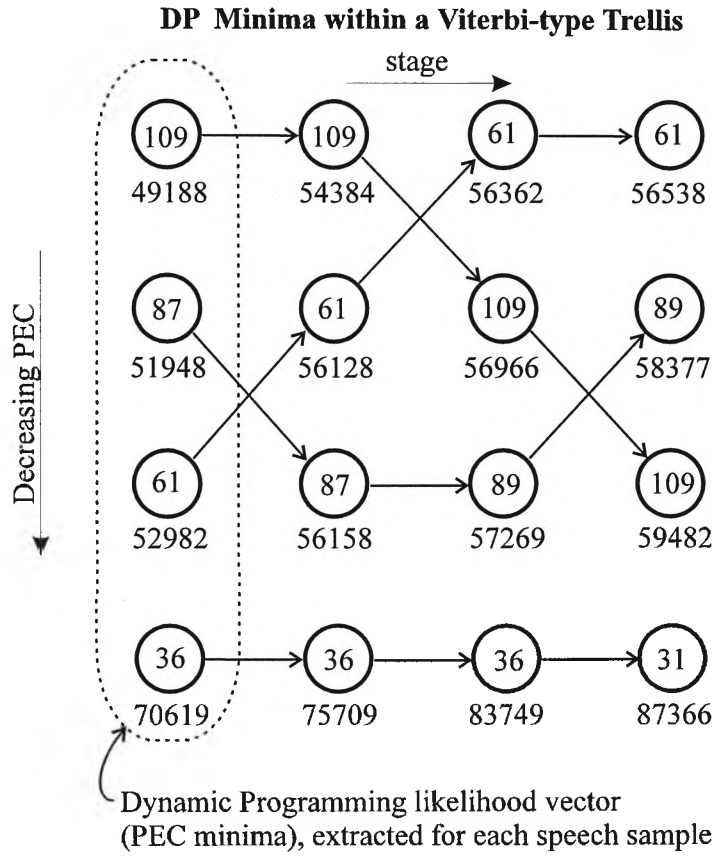

Figure 4.7 - Illustrating an array of DP derived minima as states of the Viterbi-type trellis. This array of PEC minima, extracted from frame 49 (2nd sub-frame), clearly demonstrates the large deviation between tracks 61 and 109, and the influence of Fundamental frequency harmonics.

Tracking an incorrect pitch path may also be attributed to the fact that only a select portion (depth) of the time-warped plane is retained throughout Dynamic Programming (DP) iterations [46]. The constraints (Section 4.2.3) which are imposed on the minimisation function have resulted in the unnecessary requirement of retaining previously computed data. A significant reduction in the amount of memory required to retain the DP plane is achieved when the previous DP process vectors, such as *D(i-2,j), D(i-3,j)* do not need to be retained [46], [48]. This approach results in an immediate or instantaneous value for pitch period being extracted as progression along the signal segment occurs, and not after the entire time-warped plane is traversed.

The alternative to this approach, is a 'Terminal-type' Dynamic Programming (DP) algorithm [49],[50],[52]. The Terminal-type DP method determines the optimal path after all points in the time-warped plane have been calculated. Terminal Optimisation is shown [115 p26-32] to be the Dual to the optimisation of the 'Sum-of-Stage Returns' which is implemented in the DP method presented in this thesis. Both methods are demonstrated [52] to yield the same solution, illustrating the 'Principle of Optimality'. This is, however, at odds with the requirements of a low-delay Pitch Detector, since it introduces a significant look-ahead delay cost. For this reason, DP algorithms have been used as post processors [54]. The survivor path decision process in the Viterbi Algorithm (VA) has a similar 'Principle of Optimality' [43],[44]. This provides the basis for applying the VA, in conjunction with DP-derived minima in tracking candidate pitch tracks.

4.4 Hidden Markov Model and the Viterbi Algorithm

Recently, the integration of the Hidden Markov Model (HMM) and the Viterbi Algorithm (VA) has been considered viable in speech recognition [55]. The use of HMM's in character recognition systems has revealed that the VA is particularly useful in determining the optimal path given a state symbol sequence [56]. A HMM, in conjunction with the VA, is used to select the most likely pitch track score, based on the observed state sequence. The large symbol observation space afforded by HMM is used in the DP/V algorithm to account for all possible pitch periods. In addition, a variable number of states are considered at each stage, providing the capacity to maintain smooth pitch tracks. The following sections detail these and other HMM properties applicable in modelling the speech signal, which are integral in the DP/V algorithm.

4.4.1 Symbol Observation Space

The Symbol Observation Space encompasses all permissible pitch period values which, in this implementation, range from a minimum of 20 samples to a maximum of 147 samples for speech sampled at 8kHz, accounting for a Fundamental frequency falling within the 54-400Hz band. A significant deviation from the Viterbi Algorithm [44] is the variable number of states that are considered at each stage. Excluding the first stage of each segment to be processed, the number of states can vary as progression along the symbol sequence is undertaken. The number of states is a function of the Symbol Observation Space and the Symbol Search Window (SSW), which, in this implementation, the latter is set to a minimum pitch period (20 samples [400Hz]) separation. The upper limit for the number of states is equal to the number of permitted pitch periods, which would effectively be equivalent to a table look-up.

Typically, in a Voiced speech segment, the number of valid pitch period candidates does not exceed two or three possible symbols. This accounts for both possible pitch period doubling or tripling effects. Consideration is given, however, for symbol sequences in which all likelihood candidates are absent for short durations, and therefore, a range of 3-6 states is defined (given the SSW = 20 samples, and the Symbol Observation Space = (147-20) samples a maximum of 6 states are possible).

4.4.2 Variable Number of States

An important attribute of the DP/V algorithm is the retention of states (tracks), that do not find a source state and, conversely, the propagation of the previous stage states that fail to find a destination state. This scheme allows for states to be maintained for a user defined period, after which, depending on their accumulated scores, they may either be discarded or retained for reconsideration when processing the next speech segment. To accommodate specific state transitions, the number of states can be dynamically incremented, decremented or remain constant for the duration of the current speech segment. This results in a scheme that will account for large deviations in pitch tracks, and ultimately enables the optimum (maximum accumulated score for the speech segment under test) pitch track to be selected. This is of great benefit when considering the natural occurrence of pitch period doubling, or another speech characteristic that may require the pitch period track to deviate [7] and as shown in Figure 4.8. Similarly, in the case of pitch period discontinuities as a result of signal corruption, the DP/V algorithm will retain the states, and the corresponding symbol sequences affected, for a finite duration. This is achieved by using HMM properties such as the constrained-state transitions, allowing true ergodic transitions and the Null-state transition for the propagation of symbol sequences.

52	53	32	31	28	28	28	29	66	66
32	32	144	144	125	123	66	66	30	31
108	108	108	108	70	66	123	123	123	125
143	142	82	72	100	144	102	102	103	102

Stage (speech sample) 'i'

Figure 4.8 - Array of DP-derived minima across 10 speech samples illustrating the discontinuities and deviations that can occur in speech signal (extracted from frame 49).

4.4.3 Constrained State Transitions

The Hidden Markov Model (HMM) may be extended to meet specific requirements that correctly model the signal sequence under investigation. An alternative HMM is used, which models the true ergodic system, and in addition, does not require every state to be connected directly to every other state [56]. When updating the state likelihood score, two observed exceptions must be considered. The first of these must account for short discontinuities in symbol tracks in order to maintain smooth and continuous pitch tracks. The HMM property which allows constrained 'jump' state transitions can account for the observed short discontinuity in a pitch track. This is a result of either signal corruption or, alternatively, dominant speech samples. The observed symbols may not be contiguous due to these discontinuities, hence 'jump' states correctly model such events, as illustrated in Figure 4.9.

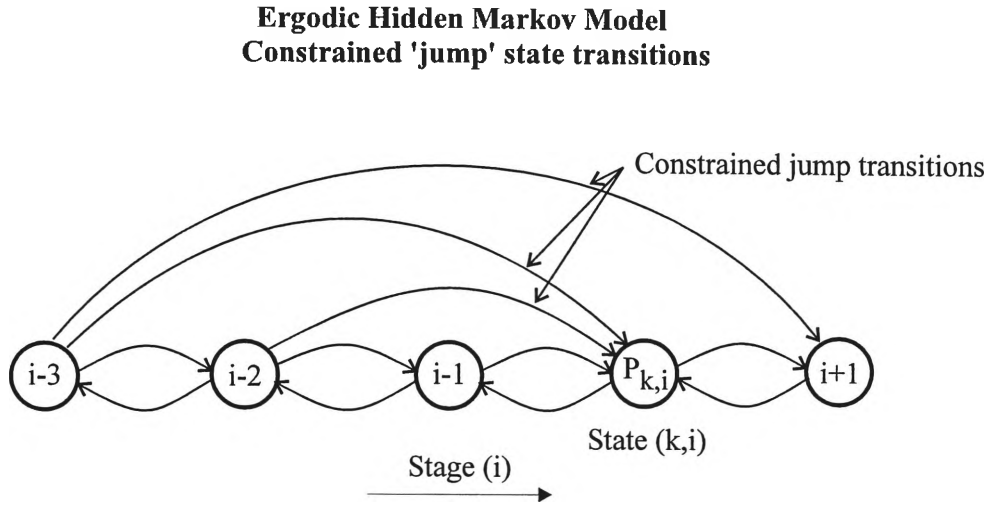

Figure 4.9 - Illustrating two constrained 'jump' state transitions for state *'k'* with symbol *'P'*, at stage *'i'*.

In terms of the Markov Model, the slope constraint imposed on the DP algorithm can also be seen as altering the Markov Model significantly, as illustrated previously in Figure 4.4. The number of source states, *D(i,j),* in the modified DP algorithm was increased which, in turn, has introduced constrained 'jump' state transitions [56].

For unconnected states, the constrained state transition permits a limited trace-back into symbol sequences in search of valid states. In this situation, a trace-back is beneficial because the current states can inherit non-directly connected state symbol sequences. This desired result achieves a smooth and continuous pitch contour. The second exception occurs when the trace-back does not find a valid source state. A new state and corresponding pitch track is then required to be generated. The DP/V algorithm allows for the evolution of new pitch tracks by evoking the HMM property of Null-State transitions.

4.4.4 Null State Transitions

When a source, or destination, state does not exist, the HMM Null-state transition property can be used to propagate symbol sequences, within the trellis, for a defined interval. This propagation can be achieved without adding to the accumulated score. The Null-state transition is illustrated by the dashed state transition in Figure 4.10.

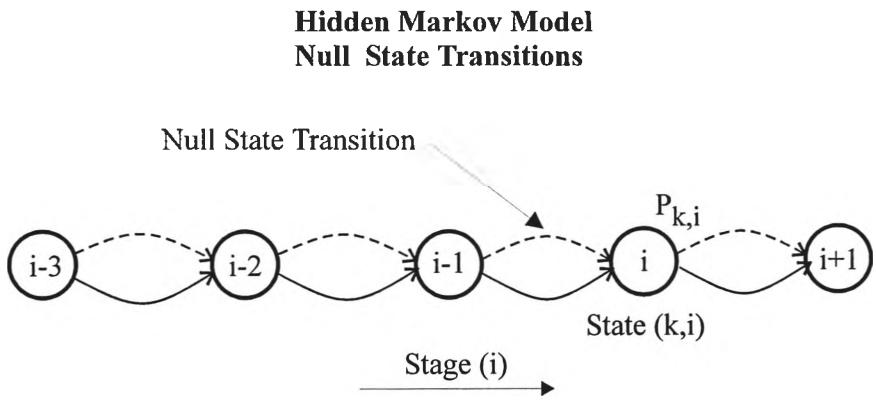

Figure 4.10 - Propagation of State $P_{k,i}$ as a result of a Null-state observation. The state symbol *'P'* remains constant as does the index *'k'*.

4.5 State Acquisition and Weighting Scheme

4.5.1 State Acquisition

The acquisition of states commences from the vector of correlations, *D(i,j),* calculated by the Dynamic Programming (DP) algorithm [47]. The DP/Viterbi states are extracted from the PEC vector *D(i,j)* (equation 4.10) at each stage *'i'*, resulting in the following state vector.

$$P_{k,i} = (P_{1,i}, P_{2,i}, P_{3,i}, ..., P_{K(i),i}) \quad (4.13)$$

where $P_{k,i}$ is the state pitch period symbol for state *'k'* and stage *'i'*.

The initial state $P_{1,i}$, is the DP-determined state with the minimum recorded PEC value for stage *'i'*. The additional states, $\{P_{2,i}, P_{3,i}...\}$ are those which, over a speech segment, are the secondary and alternative DP/V minima, resulting in additional track candidates. The selection of a DP/Viterbi state is based on the following criteria: (1) that the DP (PEC) score is a minimum; and (2) that the extracted state is unique. A duplication (similar symbol) of states can occur when the DP process results in a high correlation between adjacent speech samples. To ensure, therefore, the uniqueness in the Viterbi states, a state selection process instigating a minimum pitch period separation rule prevents duplication of states. The Symbol Search Window (Section 4.4.1) provides for a minimum pitch period separation of 20 speech samples (400Hz). The state selection process is repeated, selecting the next minimum *D(i,j),* until the required number of states are acquired. The resulting DP/Viterbi states have an associated symbol value equal to the *D(i, j)* index *'j'*, (position at which the minimum PEC occurs).

4.5.2 State Probabilities

The failure of previous Hidden Markov Models (HMM's) that are applied to speech recognition, has been due to the calculation of state transition probabilities, and the associated probability distributions that are used [56]. Two state probabilities and a transition probability are typically associated with each state in a HMM [56]. Firstly, an initial state probability is allocated to all states, and is then replaced by the state symbol observation probability. Secondly, the state transition probability is associated with the transition between states. In contrast to the HMM, the DP algorithm has been shown (Section 4.2) to use a Partial Error Criteria to determine the DP likelihood and, hence, state transitions. The DP/V algorithm presented in this thesis has removed these probabilities and, instead, has allocated symbol weights, based on the DP likelihood, to each state used in the DP/V algorithm. This alternative weighting scheme, and the corresponding likelihood update procedure, removes the calculation of the symbol observation probability and the state transition probability associated with HMM's. In addition, the resulting update of the likelihood function, based on this alternative weight scheme, removes a dependency on the signal content, whereby individual samples can cause an undesired large excursion which bias non-optimal DP pitch tracks. The result of the acquisition process is a set of Viterbi states which are then alternatively weighted.

4.5.3 Weighting Functions

The introduction of an alternative weighting scheme has removed the dependency on the actual DP-derived PEC value, $D(i,j)$ (equation 4.10), in the final pitch decision. The PEC which is calculated from the raw speech signal samples may incorrectly bias non-optimal

tracks. The PEC value is used solely for the determination and acquisition of the DP/Viterbi states. Once the required number of states have been extracted from the PEC vector *D(i,j)*, the PEC value is discarded. An alternative weighting scheme replaces the PEC value which provides the capacity to bias paths that yield smooth pitch tracks.

4.5.3.1 State Observation Weighting

Once acquired, the Viterbi states shed the DP (PEC) weight *D(i,j)* (accumulated residual error) for an alternatively-derived weight, prior to insertion into the Viterbi-type Trellis. The DP weight scheme is based on minimisation of *D(i,j)*, whereas the Viterbi states are able to use a maximisation of an alternative weight function defined as:

$$W_{k,i} = f(k), \text{ for } k = 1..K(i). \quad (4.14)$$

where $W_{k,i}$ is the symbol weight, based on the DP Likelihood, assigned to state *'k'* at time (stage) *'i'*.

This alternative weighting scheme is illustrated in Figure 4.11, where a linear weighting function has been applied to DP likelihood states. In this example (Figure 4.11) the weighting function $W_{k,i}$ has allocated an integer weight of $W_{1,1} = 3$, to the most likely state. It has then allocated a weight, $W_{2,1} = 2$, to the second candidate, a weight, $W_{3,1} = 1$, to the third state and, finally, $W_{4,1} = 0$ to the fourth state. This is repeated for each speech sample (stage) of the Viterbi-type trellis. The survivor paths (solid connections) between stages clearly illustrate the successful state transitions and, the update of the state likelihood scores.

This example clearly illustrates, after four stages, the DP derived pitch period of 121 samples with a likelihood score of '7' is not the correct pitch estimate, whereas, the DP/V generated pitch period of 54 samples, recording a maximum score of '11', is the correct pitch period.

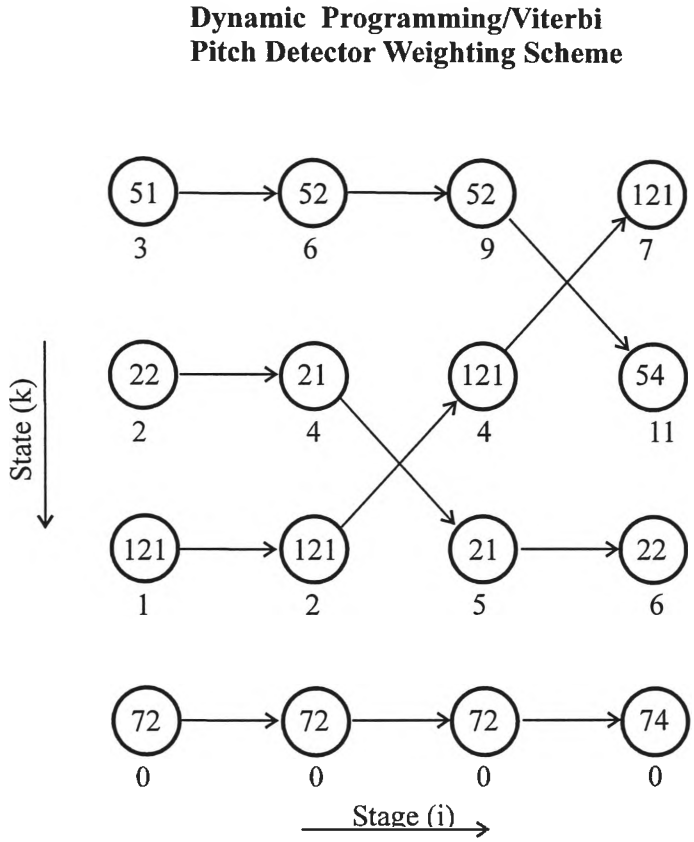

Figure 4.11 - Dynamic Programming/Viterbi Pitch Detector weighting scheme, where each state is represented by a Pitch Period (within the circle) and respective likelihood score (outside circle) replacing the DP Partial Error Criteria value.

4.5.3.2 State Transition Weighting

A Dynamic Programming/Viterbi state transition weighting provides for a similar mechanism to the Hidden Markov Model state transition probability. This additional measure, which is introduced into the state likelihood score update process, benefits cases where a high

correlation exists between state symbols. A low, or zero, difference is considered a high probability state transition, and conversely, a large symbol difference constitutes a low probability state transition. A symbol difference function, $V_{m,k}$ is defined as the error between the current symbol (pitch period estimate) and symbols from the previous stage, as is given by:

$$V_{m,k} = f\left(\left| P_{k,i} - P_{m,i-1} \right|\right), \text{ where } 1 \le m \le K(i-1). \quad (4.15)$$

where $P_{k,i}$ is the current state symbol vector and, $P_{m,i-1}$ is the previous iterations state symbol vector. This measure, in conjunction with the respective symbol observation weight, is accumulated for each current state symbol. The accumulated score is used as the likelihood function to determine the survivor path in the Viterbi-type trellis.

4.6 Dynamic Programming / Viterbi Model

4.6.1 State Likelihood

In the following model presentation two parameters, which are interchanged to convey the model description, are 'time', or 'stage', and the state 'symbol', or 'pitch period value' [56].

The Dynamic Programming/Viterbi State Likelihood vector used in this work is defined:

$$S_{k,i} = (S_{1,i}, S_{2,i}, S_{3,i}, ..., S_{K(i),i}). \quad (4.16)$$

where $S_{k,i}$ is the state likelihood score for state *'k'* at time *'i'*, and, to update this state likelihood vector, the following is used:

$$S_{k,i} = \max_{\{m\}} \left\{ S_{m,i-1} + V_{m,k} \right\} + W_{k,i} \quad \text{.....(4.17)}$$

for $k = 1..K(i)$, and $m = 1..K(i-1)$, which are the number of the current states and previous iteration stage states respectively. The update of the likelihood score (4.17) is only used if $\left| P_{k,i} - P_{m,i-1} \right| < \text{Symbol Search Window}(\text{SSW})$, which is defined as the allowable variation in individual tracks.

The DP/V algorithm must consider a variable number of states at each stage. In addition, state symbols can differ from stage to stage, as a result of the large symbol observation space [56]. This leads to a situation where not all the acquired states from the Dynamic Programming process will have a source state, or associated symbol sequence, to which they can attach themselves. To elaborate on the three state transitions that can occur, a detailed explanation of the procedure in determining the source state, and the update of the likelihood score for each state transition, is presented.

4.6.2 Current States with a Source State

For each current state $P_{k,i}$, a search of the previous stage states $P_{m,i-1}$, is undertaken to determine the optimal DP/V state transition. The DP/V optimal state transition is between states which attains the maximum accumulated weight whilst within the Symbol Search

Window(SSW). Provided that $\left|P_{k,i} - P_{m,i-1}\right| < \text{SSW}$, indicating that a source state exists, the state symbol $P_{k,i}$ is pre-appended to the symbol sequence containing symbol $P_{m,i-1}$. The likelihood score $S_{k,i}$ is then updated using (4.17). This process is continued recursively for all current states, $P_{k,i}$ (equation 4.13). This case aligns itself with the Viterbi Algorithm in the determination of the optimal source symbol and survivor path determination.

4.6.3 Current States without a Source State

If a current state does not have an associated source state a deeper search (trace-back) into all the symbol sequences, for a short duration, is performed in seeking a valid state symbol. This search is essential, as the symbol may have drifted, or may be non-existent for the current sample, and 'jump' state transitions (Section 4.4.3) are used to maintain such symbol sequences. If, however, the trace-back fails to find a valid state, this event may be the evolution of a new path, or be the result of the track being discontinuous at the current sample. In such cases the state is retained (propagated forward) for a relatively short interval for subsequent reconsideration. This is achieved by retaining respective states at their current index in subsequent iterations, effectively propagating states for reconsideration, as (4.18) illustrates:

$$S_{k,i+1} = S_{k,i} \quad \text{.....(4.18)}$$

To allow the trellis to continue with this state, a zeroing of the state path history is necessary (as the state index conflicts with an existing symbol sequence). Additionally, the respective

accumulated likelihood score is reset. With no history the new state commences to compete with existing tracks (other state likelihood scores). A trade-off is therefore required between the depth of the trellis (dominance of existing tracks), and to the rapid detection (competitive participation) of new pitch tracks. The time-variant characteristic of the pitch period, coupled with Voiced/Unvoiced transitions, dictates that new pitch tracks will require short durations (frame length order), in order to compete with existing pitch tracks. State likelihood scores are reset and biased at such a frequency (frame boundaries) to allow this to occur, permitting track selection based on accumulated scores.

4.6.4 Previous States to be Propagated

This case relates to the retention (effective propagation) of the previous iteration stage states for further consideration in subsequent stages. For each previous state, determination of a destination state is performed and, if found, no further processing is performed. If a previous state, however, does not have an associated destination state, it can be an indication of an eminent path termination, or a discontinuous path at the current sample as a result of speech signal corruption. The need to retain the path associated with this state is vital to the propagation of its symbol sequence. In propagating this state, an additional state is generated using the 'Null-state' transition (Section 4.4.4), and the symbol sequence can, therefore, be maintained for a finite duration. If, after this period, no improvement in the likelihood score is observed, no further consideration will be given to this state when processing the next speech segment. In a similar process to the 'Current States with a Source State' (Section 4.6.2), all of the previous stage states ($m = 1 .. K(i-1)$), are tested for discontinuities in tracks, using the SSW. If a track is discontinuous, then an additional state is generated for the current stage using:

$$S_{K+1,i} = S_{m,i-1} \quad \text{.....(4.19)}$$

This effectively propagates the previous state to allow reconsideration of such tracks in subsequent iterations. The variable number of states, *K(i)* to be processed at each stage, comprises current states and propagated (additional) states from the previous iteration. Attributes of the model presented include effectively propagating current states and previous iteration states, if either source or destination states respectively do not exist. The propagation is only carried out for a finite duration (segment length), after which, if the accumulated likelihood score has not increased (relative to other candidate states), it is discarded from the track list prior to processing the next speech segment. The Dynamic Programming/Viterbi (DP/V) model presented has introduced a variable number of states to enable pitch tracks to be maintained within a Viterbi-type trellis. Initially, the number of states is fixed (initial sample) for each speech segment to undergo the DP process. The variation occurs as progression along the input sequence (sample-by-sample) occurs.

4.7 The DP/Viterbi Pitch Detection Algorithm

The iterative algorithm can be divided into four basic steps:

Step 1: The speech frame is divided into 40 sample (5ms) segments (sub-frames). For each sample in the segment, a vector of correlations with surrounding samples, within realistic pitch period limits (20-147 samples), is computed. Figure 4.12 illustrates the respective speech segments which undergo the Dynamic Programming process.

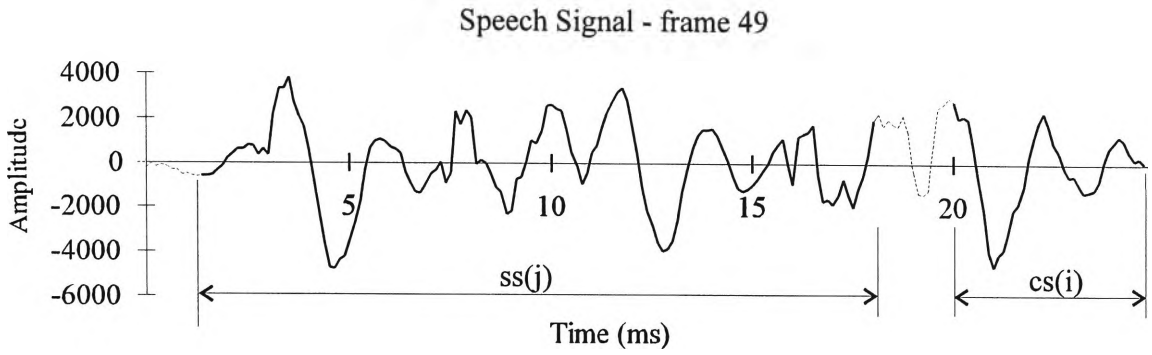

Figure 4.12 - Speech sub-frame cs(i) [20-25ms] to be DP matched with preceding speech samples ss(j) [1.5-17.5ms], accounting for the valid pitch period interval between 20-147 speech samples.

Step 2: Based on the correlation vector, the DP process is used to form a 'likelihood vector', which incorporates information from the surrounding samples in the two-dimensional DP time-warped plane D(i,j) (it should be noted that correlation maxima are now described as Partial Error Criteria minima in the DP plane).

Step 3: The 'likelihood vector' is then used as an input to a Viterbi-type process. From the vector a finite set of minima are extracted, weighted, and then input into a Viterbi trellis. This leads to a set of 'weighted' possible pitch tracks, expressed as paths in the trellis. In reference to Figures 4.13a-b, example state transitions are illustrated, whereby, a linear weighting scheme was used to clearly illustrate the update of likelihood scores. Attributes, such as: (1) the maintenance of candidate paths (survivor paths); (2) the addition of new states; and (3) state removal, are observed. The large excursions of the DP-derived track (state 1) and secondary tracks (states 2, 3, 4...) are also clearly illustrated in Figure 4.13a. At the 10th stage, the second state attains the highest score '39', recording a pitch period value of '54' samples, clearly illustrating that a large deviation has occurred. The evolution of a new track by the insertion of symbol '21', evoking the Null-state transition, occurs at stage 3.

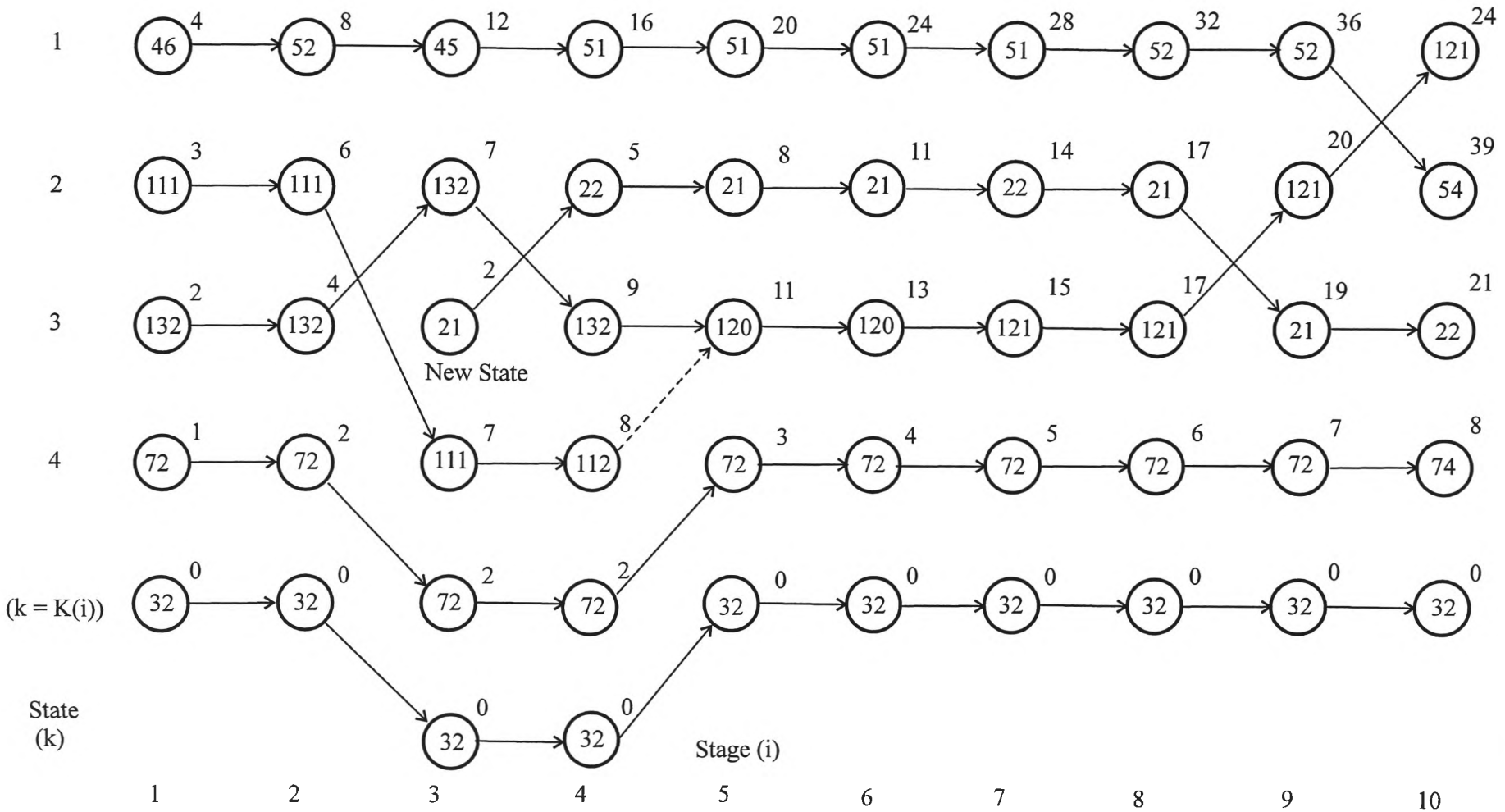

Figure 4.13a - Trellis Diagram (Frame 49) illustrating the additional state generated at stage 3 and the removal of the state with symbol value 112 in the fifth stage. The progress of symbol 121 in stages 8, 9 and 10 illustrates the alteration in the likelihood track order.

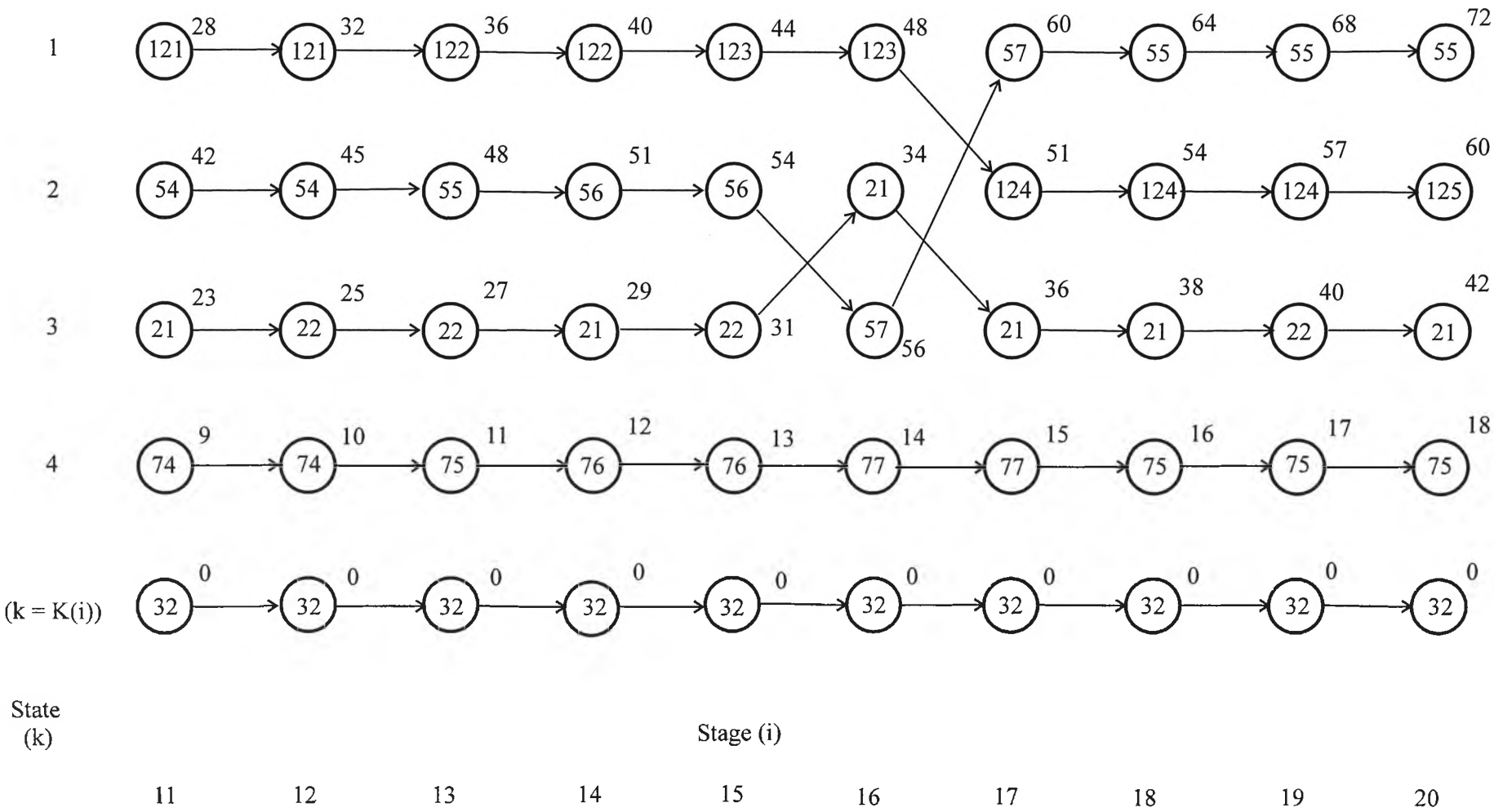

Figure 4.13b - Trellis Diagram (Frame 49) illustrating the alteration in the likelihood track order at stages 15, 16 and 17. The DP/V pitch track selected is that with a symbol value of 55 with the maximum score of 72 after 20 stages.

The broken (dashed) paths distinguish a survivor path from a non-survivor path as shown at stages 4 and 5, resulting in the subsequent state elimination. Continuing until the 20th stage, the optimal path is that which records a pitch value of '55' with an accumulated score of '72'.

Step 4: While a pitch decision can be derived from the trellis at every sample interval, for reasons of complexity the pitch decision is restricted to 5ms intervals. The selection of the optimal path is based on the maximum-likelihood scores, of prospective paths in the trellis, attained after a predefined number of stages, which, in the current implementation, a pitch decision is extracted after 20 stages (5ms). The associated 'likelihood vector' of accumulated scores is reset on frame boundaries to allow rapid pitch detection. The number of states (tracks) to be retained at the completion of a speech frame is fixed at four states. This allows the number of states to increase, or decrease, within the frame interval.

4.8 Open Loop Operation and Median Filter

The DP/V algorithm currently functions as an open-loop system. No pitch doubling or tripling checks are performed. Pitch doubles and triples are still observed, however a 5-point median filter [57] reduces their occurrence producing smooth pitch tracks. In the event of a pitch double, current Prototype Waveform techniques are able to account for such an occurrence, given that the estimated pitch period is a harmonic of the true pitch period [58]. The overall outcome of the above attributes is the provision of a basis for the detection of the pitch period at a significant reduced look-ahead delay than that afforded by other algorithms that require in the order of a entire frame look-ahead, to achieve satisfactory robustness and accuracy.

4.9 DP/V Pitch Detector Results

TABLE 4.1

Results for the Dynamic Programming/Viterbi Pitch Detector applied to a 2024 frame speech data base, of which 1079 are considered voiced

Error Classification	Count	Average (samples)	Std. Dev. (samples)
Voiced frames with no errors	292	0	N/A
Voiced frames with Fine Errors [≤ 8 samples (≤1ms)]	690	2	2
Voiced frames with Gross Errors [> 8 samples (>1ms)]	97	22	16
Total voiced frames	1079	3	9

The DP/V Pitch detector has been tested across a wide speaker database comprising 20 male and female 'British' English spoken sentences [35], producing accurate results. The results are summarised in Table 4.1 with corresponding DP/V-generated pitch profiles versus manually-derived profiles in Appendix D. These figures plot the results on a frame-by-frame basis, hence, a pitch value every 200 sample (25ms) frame is recorded. Even though the pitch period estimate is determined every 40 samples (5ms), the pitch value recorded 15ms into the frame is that which is recorded for the overall frame, so that a comparison with other pitch

pitch detectors presented can be performed. Due to the requirement of including a 5-point median filter, a delay of 12.5ms (5x5ms/2) has been incurred (the input to the median smoother is the pitch value at intervals of 5ms). The results tabled in Table 4.1 indicate that the small percentage of gross errors recorded have contributed significantly to the overall average error for all voiced frames. These Gross errors in some instances are attributed to the manual pitch profile, whereby solitary pitch excursions were permitted, as no smoothing was undertaken for this reference profile. Overall it is considered that, phenomena such as pitch period doubling is still observed but to a lesser extent.

4.10 Conclusions

When using Dynamic Programming techniques observations of the output have revealed that large deviations in pitch tracks do occur. However, when more than one DP-derived minimum is retained, then a track can be observed to exist across this array of minima. This observation, therefore, provides the necessary information to maintain a set of smooth pitch tracks via a Viterbi-type algorithm. Additionally, the introduction of an alternative weighting scheme assists to maintain candidate pitch tracks, reducing the occurrence of non-optimal track selection.

The algorithm excels in, primarily, its ability to avoid consideration of unwanted signal samples that may be corrupted by noise. This is achieved via Dynamic Programming, optimising on a sample-by-sample basis. The key concept underlying this fact is that, when calculating the pitch period, the delay between signal samples need not be constant. Secondly, the introduction of a variable-state Viterbi-type trellis, enabling the addition of

new states and therefore maintaining likelihood paths, avoids the selection of non-optimal pitch tracks. This is of great benefit when considering the natural occurrence of pitch doubling or other speech characteristics that results in the pitch period track to deviate significantly. The advantage of the Viterbi-type trellis is that redundant, and potentially misleading, information from the DP algorithm can be removed by using path likelihood. This is a result of the algorithm considering sample-by-sample pitch behaviour, while utilising the concepts of natural variation.

The overall scheme provides the basis for detection of the pitch period of speech, while incurring a lower delay, dispensing with the significant look-ahead delay of the autocorrelation-based Pitch Detectors. When decisions are restricted to, for example, 5ms intervals, the algorithm complexity is not substantially greater than other pitch detection mechanisms [7]. These results indicate that DP/V-based Pitch Detection is promising for reduced delay speech coding.

CHAPTER 5

CONCLUSION

5.1 Comparative Results and Summary

A summary of the results of the four algorithms implemented in this thesis are presented in Table 5.1, from which the Pitch Detection performance of each algorithm can be compared. The results tabulate, on a frame-by-frame basis, the Fine and Gross pitch errors and respective Mean and Standard-Deviation, for each algorithm. To permit this comparison, only one pitch value is recorded per 25ms frame, even thought the PWPD and the DP/V algorithms can generate a pitch estimate at 5ms intervals. For both of these latter algorithms, only the pitch period at the centre of the frame is recorded.

Referring to Table 5.1 for the category of 'No Errors' (the generated pitch period equals the reference pitch period), the DP/V algorithm has recorded the highest score of 292 frames with, from a total of 1079 voiced speech frames. This is followed by the GCI and the PWPD recording similar scores of 239 and 241 respectively, for this error category. Within the 'Fine Error' category (errors $\leq$ 1ms), the PWPD algorithm has recorded the highest score of 763 frames with 'Fine Error'. The SIFT and the GCI algorithms follow with scores of 725 and 691 respectively. The DP/V algorithm recorded the lowest score of 690 Fine Errors. For the 'Gross Error' category, the lower the score, the greater is the performance of the algorithm. Within this category, the PWPD recorded the lowest score of 75 Gross Errors, followed by the SIFT and DP/V algorithms with scores of 95 and 97 respectively. The GCI recorded the poorest performance, almost twice as many Gross Errors than produced by the PWPD.

Table 5.1

Summary of the results from the four algorithms that were implemented in this thesis.

Error Classification (Voiced frames only)		SIFT	GCI	PWPD	DP/V
No Errors	Count	259	239	241	292
	Mean	0	0	0	0
	Std.Dev.	N/A	N/A	N/A	N/A
Fine Errors	Count	725	691	763	690
(≤ 8 samples)	Mean†	2	2	2	2
	Std.Dev.†	1	2	1	2
Gross Errors	Count	95	149	75	97
(> 8 samples)	Mean†	32	28	42	22
	Std.Dev. †	21	21	34	16
Total Errors	Count	1079	1079	1079	1079
	Mean†	4	5	4	3
	Std.Dev. †	12	14	16	9

(Note: † mean and standard deviation units are samples of speech sampled @8kHz)

The calculation of the 'Total Error' category has revealed that the DP/V algorithm has recorded the lowest Mean and Standard Deviation overall. The performance of the PWPD is affected by a relatively high standard-deviation which is attributed to the high occurrence of Gross Errors at the Voiced-to-Unvoiced transitions and vice-versa. These transition segments are relatively short in duration (<frame length), however, this is sufficient to generate a false autocorrelation estimate.

5.2 Conclusion

This thesis has presented the results of research into Low-Delay Pitch Detection, specifically targeted for low-bandwidth (2400bps) speech coding algorithms for use across telecommunications networks. The detection of the pitch period at Low-Delay is an important attribute required by a successful low bit-rate speech coder. The accurate determination of the pitch period is considered vital if the quality of the reconstructed speech is to match, or supersede, its higher bit-rate counterpart. The results in achieving an improved pitch period at Low-Delay would significantly enhance the quality of decoded speech, approaching that of the original speech. Other applications which make use of coded speech, such as Speaker Verification, Word/Digit Recognition Systems, and sophisticated speech encryption algorithms will also benefit significantly from improved Low-Delay Pitch Detection.

The impediment to a successful Pitch Detector suitable for Low-Delay speech coders is the accurate detection of the pitch period at a low algorithmic delay. The Formant (resonant) frequencies generated by the Vocal and Nasal Tracts are characteristic of voiced speech and have also been illustrated in this thesis to vary with time. The Short-term Linear Prediction (LP) technique offers an efficient method to determine the Vocal Tract model, whereby, voiced speech segments are efficiently represented by LP coefficients over short 20ms speech frames. The pitch period, which is a Long-Term speech component, is an additional characteristic of voiced speech that is not efficiently modelled by the Formant filter. This Long-Term component models the Fundamental frequency by a periodic pulse train, observable, in most instances, from the Linear Prediction residual. In the determination of this Long-Term (Pitch) component this thesis has presented the results of four prospective Pitch Detection algorithms.

The Literature Review presented in Chapter 2 commenced with an early Real-time Pitch Detector based on the Auto Correlation Function. This algorithm implemented an effective Pitch Detector by non-linearly pre-processing the speech signal to overcome the ambiguity caused by secondary signal components which are detrimental to the detection of the Fundamental component. These unwanted signal components include the lower Formant frequencies, and the harmonics of the Fundamental component. The Simplified Inverse Filter Tracking (SIFT) Algorithm is then implemented presenting an algorithm which removed unwanted signal components by using Low-Order (4th Order) Linear Predictive filtering. This method was based on removing unwanted signal components sufficiently whilst retaining significant signal periodicities in order to enhance pitch epochs to aid in the detection of the Fundamental component. The SIFT algorithm provided a good baseline for the Pitch Detection work undertaken in this thesis.

The Glottal Closure Interval (GCI) Pitch Detector, also implemented in this thesis, provided an second alternative approach to Pitch Detection, whereby, the signal periodicity was emphasised by making use of the Vocal Tract impulse response in conjunction with the Hilbert Transformation. The GCI technique considered the non-causal speech characteristics exhibited predominantly by the Nasal Tract. The GCI detection method proposed the enhancement of the glottal instances by using an appropriate 'Selection Signal' to model the periodic pitch pulse train which was then correlated with the impulse response of the Vocal Tract. The results presented clearly demonstrate the capability of both the SIFT and GCI algorithms in generating accurate pitch tracks. In striving for a further algorithmic delay reduction, two alternative techniques, the Prototype Waveform Pitch Detector and Dynamic Programming/Viterbi Pitch Detector, were developed for Low-Delay speech coding. These

techniques were developed in conjunction with the requirement to achieve the required robustness and accuracy.

As a result of the Pitch Detection research undertaken in this thesis a selection of current time-domain Pitch Detection algorithms were revisited. The SIFT and GCI techniques were selected on the capacity to generate an accurate pitch estimate, while incurring a relatively low (20-25ms) algorithmic delay. Their respective algorithmic delays (an entire frame), however, do not meet the requirements for low-delay speech coders and, are therefore, not preferred. The algorithmic delay in detecting the pitch period has been demonstrated to be one of most important evaluation criterion. The delays incurred by both the PWPD and DP/V algorithms go a significant way in achieving a targeted total overall speech coding one-way delay of 90 ms.

The Prototype Waveform Pitch Detection (PWPD) algorithm which was presented in Chapter 3 is based on the Short-Term Composite Auto Correlation Function. Within the paradigm of Prototype Waveforms, the PWPD algorithm achieves accurate pitch tracks, enhancing the extraction of prototypes (pitch period in length) from the speech (residual) signal. This is achieved by tracking the constituent autocorrelations within a speech frame, therefore maximising the detection of the pitch period variation. The algorithm is capable of maintaining smooth pitch tracks at a significantly lower look-ahead delay than that required by alternative autocorrelation techniques. This results in the pitch period being extracted at frequent (5ms) intervals if required. These Prototypes are then parameterized using the Discrete Fourier Transform. This parameterization, therefore, permits linear interpolation of the prototypes, significantly reducing the required transmission bandwidth.

The second new algorithm, 'Dynamic Programming/Viterbi (DP/V) Pitch Detection' was presented in Chapter 4. This algorithm is based on the 'Principle of Optimality' and is implemented using Dynamic Programming (DP), with a substantial extension incorporating a Viterbi-type trellis. The DP strengths lie in its non-linear signal matching capabilities. A deficiency arises, however, when incorrect (non-optimal) pitch tracks result. To overcome this situation the DP/V algorithm proposed maintains multiple (candidate) tracks from which an optimal track, based on the track likelihood score, is selected. This is achieved by retaining secondary DP minima, which have been shown to be valid pitch candidates, within a Viterbi-type trellis. Consequently, the significant look-ahead delay, incurred by previous DP based implementations, to allow for corrections to the final track, is eliminated. This delay excludes DP based Pitch Detectors from consideration for use in Low-Delay speech coding algorithms. The incorporation of a Viterbi-type trellis in the DP/V algorithm has provided a robust open-loop scheme, reducing the ambiguity in the pitch period detection while achieving smooth pitch tracks at low-delay. In achieving this, the Viterbi trellis survivor path determination process has modelled state transitions on an extended Hidden Markov Model (HMM). The HMM can account for pitch track discontinuities by providing mechanisms whereby candidate pitch tracks can be propagated within the trellis maintaining smooth pitch tracks. The results of the DP/V algorithm presented in this thesis have demonstrated a reduction in the occurrence of pitch doubling or tripling.

5.3 Contributions

To summarise the contributions discussed in this thesis are:

1.) A review of early Pitch Detection techniques highlighting the impediments to detecting the Fundamental frequency of the speech signal was presented. The removal of the Fundamental component from the speech or residual signal is to the detriment of Pitch Detectors which rely on its existence and, therefore, not robust. Techniques which emphasise the signal periodicity and, are not dependant on the presence of the Fundamental component, such as the GCI, are preferred.

2.) The Prototype Waveform Pitch Detector (Chapter 3) was presented which is based on an improved 'Composite' Auto Correlation Function. This improved method is able to detect the pitch period at a significantly lower (approximately 30%) look-ahead algorithmic delay than existing autocorrelation methods that require a whole frame look-ahead. This is achieved within the paradigm of Prototype Waveforms by tracking the constituent intra-frame autocorrelations and, therefore, providing the algorithm with the capacity to detect the variation in pitch period rapidly.

3.) A Dynamic Programming/Viterbi Pitch Detector is presented Chapter 4 which extends the Dynamic Programming (DP) method by introducing a Viterbi-type Trellis to maintain candidate pitch tracks. By determining secondary and subsequent DP-derived minima and inserting them into a Viterbi-type trellis expressed as paths, the selection of non-optimal tracks can be avoided.

5.4 Future Work

The Dynamic Programming/Viterbi algorithm has produced an overall better performance, with respect to accuracy, than the preceding Pitch Detectors (SIFT, GCI and PWPD) and, at a significantly lower algorithmic delay than the frame based SIFT and GCI techniques. Future work would predominantly then concentrate on the DP/V algorithm. Although good results from the DP/V algorithm have been produced, further work is suggested in the area of the DP/V weighting functions. The weighting functions for both state observation and state transition are targeted for improvement. Currently a linear weighting scheme is applied to each state observation, irrespective of the accumulated state score. It is suggested, therefore, that an adaptable state observation weight could improve results. Additionally, the state transition weight, which is currently not implemented, be also targeted for future work. It is envisaged that this work would lead to an even further reduction in the occurrence of pitch doubles and triples, and possibly lead to the removal of the Median filter altogether. The open-loop operation of the DP/V algorithm allows for large excursions in pitch track to occur, accounting for natural pitch variation.

Bibliography

[1] M. A. Kohler, L. M. Supplee, T. E. Tremain, *Progress Towards a New Government Standard 2400 bps Voice Coder*. IEEE Proc. Int. Conf. Acoust. Speech Signal Process. May 1995 p.488-491.

[2] B. S. Atal, *New Directions in Low Bit Rate Speech Coding*. Conference Record of the Twenty-Fifth Asilomar Conference on Signals, Systems and Computers p.933-934 Vol. 2., 4-6 Nov. 1991. Pacific Grove, CA, USA.

[3] I. S. Burnett, G. J. Bradley, *New Techniques for Multi-Prototype Waveform Coding at 2.84kb/s*, IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process., May 1995.

[4] W. B. Kleijn, *Encoding Speech Using Prototype Waveforms*, IEEE Trans. Speech and Audio Processing, Vol. 1, No.4, pp. 386-399, Oct. 1993.

[5] W. B. Kleijn, J. Haagen, *Transformation and Decomposition of the Speech Signal for Coding*. IEEE Signal Processing Letters Sep 1994.

[6] L. R. Rabiner, R. W. Schafer, *Digital Processing of Speech Signals*. 1978 Prentice-Hall, Signal Processing Series.

[7] W. Hess, *Pitch Determination of Speech Signals*, Springer-Verlag 1983.

[8] J. G. Proakis, D. G. Manolakis, *Digital Signal Processing, Principles, Algorithms and Applications* 2^e, Macmillan Publishing Company 1992.

[9] J. H. Chen, R.V Cox. *'The creation and evolution of the 16 kbit/s LD-CELP: From concept to standard,'* Speech Communication, Vol. 12, No. 2 June 1993.

[10] I. S. Burnett, P. M. B Gambino. *Pitch Detection based on Prototype Waveforms*. IEEE Proc. of the Fourth International Symposium on Signal Proc. and its Applications. Gold Coast Australia, 26th-30th Aug. 1996.

[11] Y. Medan, E. Yair, D. Chazan, *Super Resolution Pitch Determination of Speech Signals*. IEEE Trans. Signal Processing Vol. 39 No.1, Jan 1991.

[12] C. A. McGonegal, L. R. Rabiner, A. E. Rosenberg., *"A Subjective Evaluation of Pitch Detection Methods using LPC Synthesised Speech"* IEEE Trans. Acoustics, Speech and Signal Processing June 1977.

[13] J. J. Dubnowski, R. W. Schafer, L. R. Rabiner, *Real Time Digital Hardware Pitch Detector*, IEEE Trans. Acoustics, Speech and Signal Processing, Vol. 24, No.1, 1976.

[14] M. M. Sondhi. *New Methods of Pitch Extraction.* IEEE Trans. on Audio and Electro Acoustics Vol. AU-16, No.2 June 1968.

[15] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, C. A. McGonegal., *"A Comparative Performance Study of Several Pitch Detection Algorithms"* IEEE Trans. Acoustics, Speech and Signal Processing Oct 1976.

[16] J. D. Markel., *"The SIFT Algorithm for Fundamental Frequency Estimation"*, IEEE Trans. Audio and Electro Acoustics Vol AU-20 , No. 5, Dec., 1972.

[17] M. J. Ross, H. L. Shaffer, A. Cohen, R. Freudberg, H. J. Manley, *Average Magnitude Difference Function Pitch Extractor.* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-22, No. 5, Oct 1974.

[18] *DP-based Determination of F0 Contours from Speech Signals*, A. Kiebling, R. Kompe, H. Nieman, E. Nöth. IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process., Vol II 17-20, 1992.

[19] J. A. Moorer, *The Optimum Comb Method of Pitch Period Analysis of Continuous Digitized Speech.* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-22, No. 5, Oct 1974.

[20] S. Seneff , *"Real-Time Harmonic Pitch Detector "* IEEE Trans. Acoustics, Speech

and Signal Processing Vol ASSP-26 No. 4, Aug. 1978.

[21] K. Srinivasan, A. Gersho. *Voice Activity Dectection for Cellular Networks*. IEEE Workshop on Speech Coding for Telecommunications Oct., 1993.

[22] L. J. Siegel, A. C. Bessy. *Voiced / Unvoiced / Mixed Excitation Classification of Speech*. IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-30, No.3, June 1982.

[23] Y. Qi, B. R. Hunt. *Voiced-Unvoiced-Silence Classifications of Speech using Hybrid Features and a Network Classifier*. IEEE Trans. Speech and Audio Processing Vol. 1, NO. 2, April 1993.

[24] L. R. Rabiner. *"On The Use Of Autocorrelation Analysis For Pitch Detection"*, IEEE Trans. Acoustics, Speech and Signal Processing Feb., 1977.

[25] J. D. Wise, J. R. Caprio, T. W. Parkes, *"Maximum-Likelihood Pitch Estimation "* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-24, No.5, Oct. 1976.

[26] D. H. Friedman, *"Pseudo- Maximum-Likelihood Speech Pitch Extraction "* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-25, No.3, Jun 1977.

[27] D. H. Friedman, *"Multidimensional Pseudo- Maximum- Likelihood Pitch Estimation "* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-26, No.3, Jun 1978.

[28] A. Gersho, *Advances in Speech and Audio Compression*. Proceedings of the IEEE, Vol. 82, No. 6, June 1994.

[29] J. S Marques, I. M. Trancoso, J. M. Tribolet, L. B. Almeida. *Improved Pitch Prediction with Fractional Delays in CELP coding*. IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process., pp 665-668, 1990.

[30] P. Kroon, B. S. Atal. *On the use of Pitch Predictors with High Temporal Resolution.*

IEEE Trans. Signal Processing Vol. 39 No. 3, March 1991.

[31] W. B. Kleijn, R. P. Ramachandran, P. Kroon, *Interpolation of Pitch Predictor Parameters in Analysis-by-Synthesis Speech Coders*. IEEE Trans. Speech and Audio Processing, Vol 2, No. 1 Part 1, Jan. 1994.

[32] W. B. Kleijn, W. Granzow. *Methods for Waveform Interpolation in Speech Coding*. Digital Signal Processing. I - 215-230 1991.

[33] Y. M. Chang, D. O'Shaughnessy., *"Automatic and Reliable Estimation of Glottal Instant and Period"* IEEE Trans. Acoustics, Speech and Signal Processing Vol. 37, No. 2, Dec. 1989.

[34] J. D. Markel, A. H. Gray Jr., *"A Linear Prediction Vocoder Simulation Based upon the Autocorrelation Method"* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-22, No. 2, Apr., 1974.

[35] *IEEE Recommended Practice for Speech Quality Measurements*. IEEE Trans. on Audio and ElectroAcoustics Vol. AU-17, No. 3, Sep. 1969.

[36] T. V. Ananthapadmanabha, B. Yegnanarayana, *"Epoch Extraction from Linear Predication Residual for identification of Closed Glottis Interval"* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-27, No. 4, Aug 1979.

[37] T. V. Ananthapadmanabha, B. Yegnanarayana, *"Epoch Extraction of Voiced Speech"* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-23, No. 6, Dec 1975.

[38] B. S. Atal, L. R. Rabiner., *"A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to speech Recognition"* IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-24, No. 3, June 1976.

[39] W. B. Kleijn, P. Kroon, L. Cellario, D. Sereno, *A 5.85kb/s CELP Algorithm for Cellular Applications*, IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process.,

Vol. II, pp596-599, 1993.

[40] Y. Shoham, *High Quality Speech Coding at 2.4 kbps to 4.0 kbps based on Time-Frequency Interpolation*. IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process., Vol. II pp.167-170, 1993.

[41] Y. Shoham, *Speech Coding at 2.4 kbps and below via Time-Frequency Interpolation*. IEEE Workshop on Speech Coding for Telecommunications. Oct. 1993

[42] P. Vary, K. Hellwig, R. Hofman, R. J. Sluyter, C. Galand, M. Rosso. *Speech Codec for the European Mobile Radio System*. IEEE Proc. Int. Conf. Acoust., Speech, and Signal Process., pp.227 - pp230, 1988.

[43] R. Bellman, *Adaptive Control Processes: A Guided Tour*. Princeton University Press. 1961.

[44] A. J. Viterbi, *"Error Bounds for convolutional Codes and an Asymptotically Optimum Decoding Algorithm"*, IEEE Trans. on Information Theory, April 1967.

[45] P. M. B Gambino, I. S. Burnett. *Low Delay Pitch Detection using Dynamic-Programming/Viterbi Techniques*. IEEE Proc. of the Fourth International Symposium on Signal Proc. and its Applications. Gold Coast Australia, 26th-30th Aug. 1996.

[46] H. Ney, *"A time warping approach to fundamental period estimation"* IEEE Trans. Systems Man and Cybernetics, May/June 1982, p383-388.

[47] H. Sakoe, S. Chiba, *"Dynamic Programming Algorithm Optimisation for Spoken Word Recognition"* IEEE Trans. Acoust. Speech Signal Processing. Feb 1978.

[48] G. White, *"Dynamic Programming, the Viterbi Algorithm, and Low Cost Speech Recognition"* IEEE Proc. Int. Conf. Acoust. Speech Sign Process. Apr 1978 p.413-417 .

[49] R. Bellman, R. Dreyfus, *Applied Dynamic Programming*. Princeton University Press. 1962.

[50] R. Dreyfus, *Dynamic Programming and the Calculus of Variations*. Academic Press. 1965.

[51] Kaufmann, Croon, *Dynamic Programming*. Academic Press, 1967.

[52] Nemhauser, *Introduction to Dynamic Programming*. John Wiley & Sons 1966.

[53] H. F. Silverman, D. P. Morgan, *"The Application of Dynamic Programming to Connected Speech Recognition"* IEEE Acoustics Speech and Signal Processing Magazine Jul. 1990.

[54] D. Talkin, *A Robust Algorithm for Pitch Tracking (RAPT),* Speech Coding and Synthesis, Edited by W.B Kleijn and K.K Paliwal, Elsevier Science B.V. 1995.

[55] Hui-Ling Lou, *Implementing the Viterbi Algorithm*, IEEE Signal Processing Magazine, Sept. 1995.

[56] L. R. Rabiner, *"A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition"*, IEEE Proceedings Feb. 1989

[57] L. R. Rabiner, M. R. Sambur, C. E. Schmidt, *Applications of a Nonlinear Smoothing Algorithm to Speech Processing*, IEEE Trans. Acoustics, Speech and Signal Processing Vol. ASSP-23, No.6, Dec 1975.

[58] Speech Coding and Synthesis, Edited by W.B Kleijn and K.K Paliwal, Elsevier Science B.V. 1995.

APPENDIX A

Generated Pitch Profiles

using

Simplified Inverse Filter Tracking

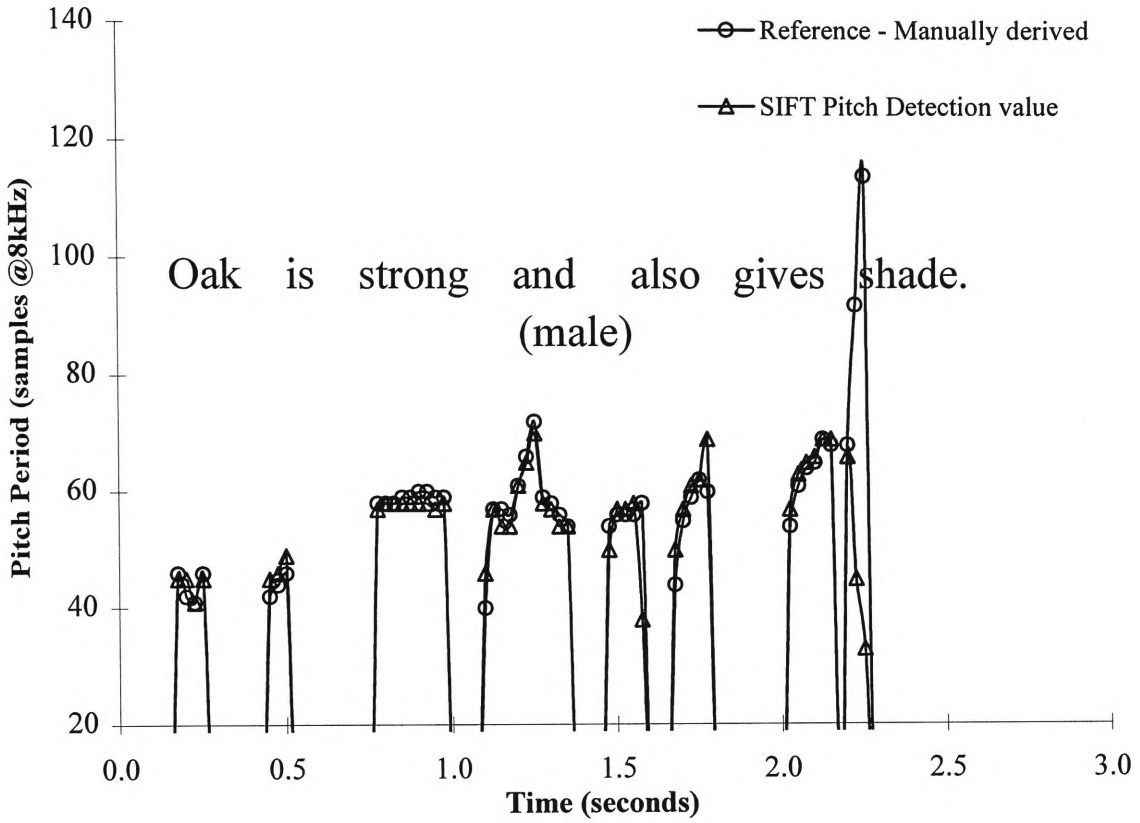

Figure A1 - SIFT Pitch Detector generated Pitch Profile

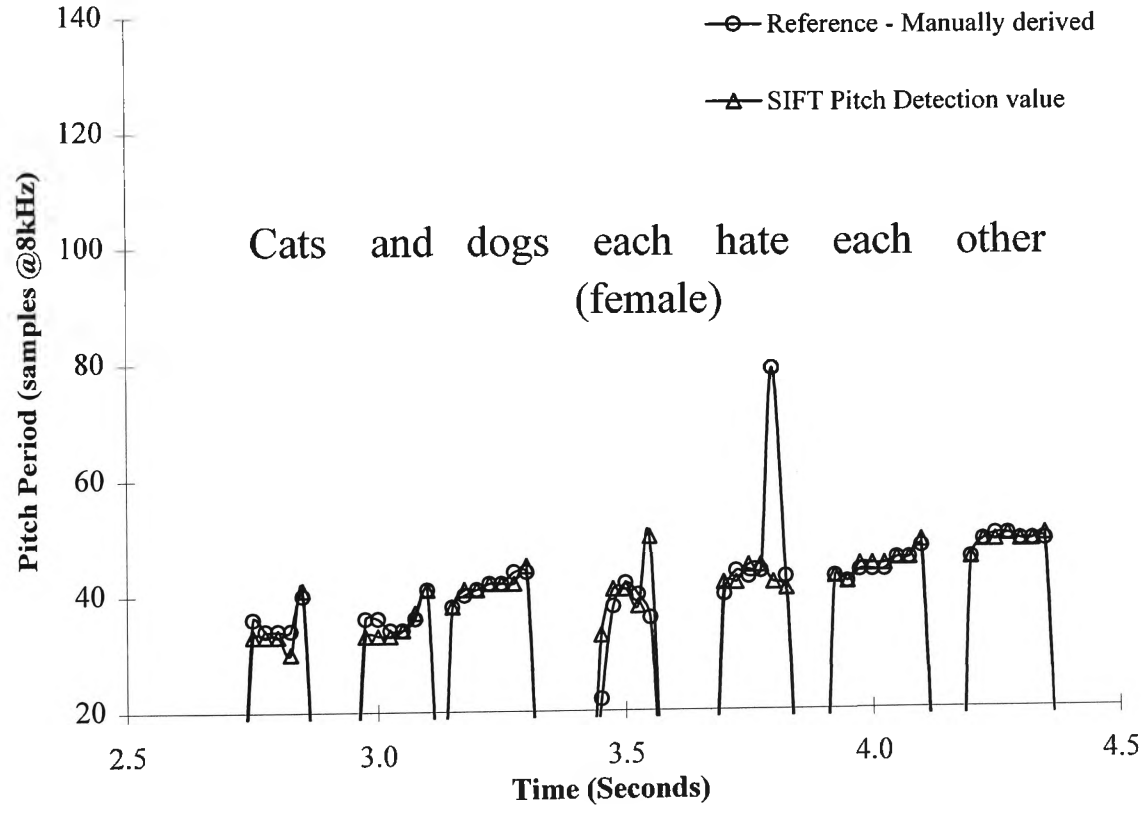

Figure A2 - SIFT Pitch Detector generated Pitch Profile

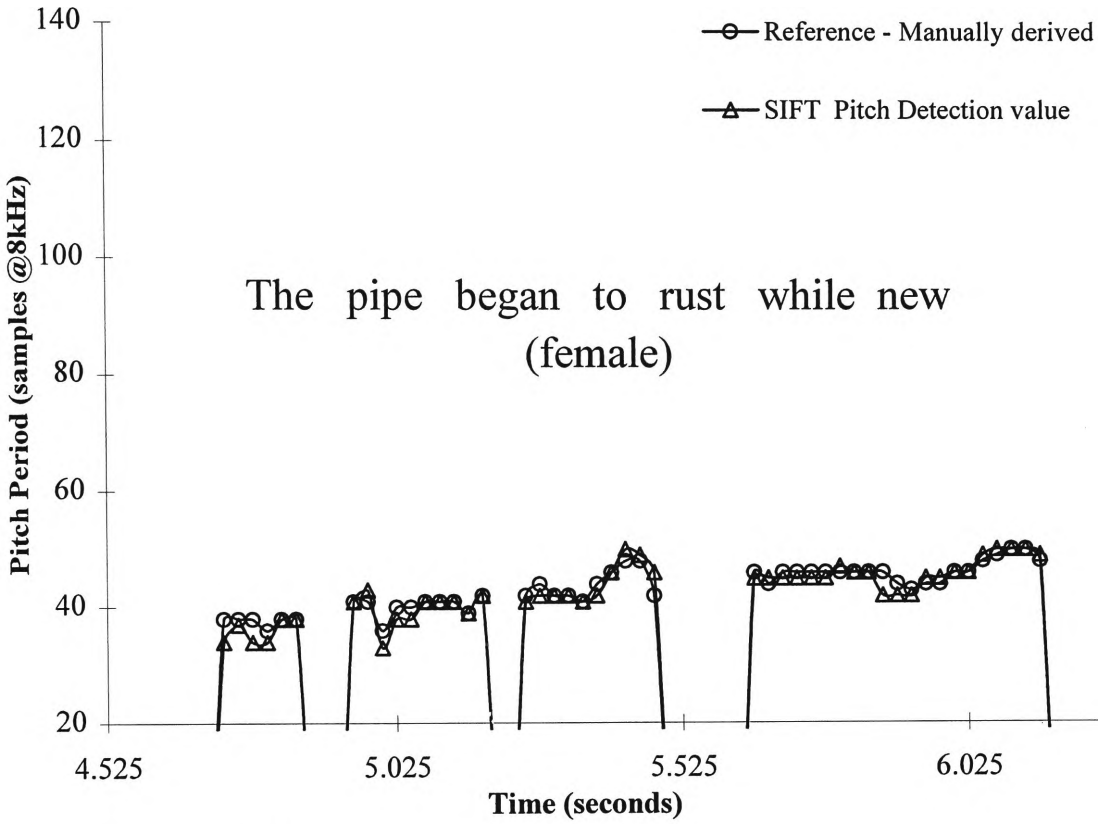

Figure A3 - SIFT Pitch Detector generated Pitch Profile

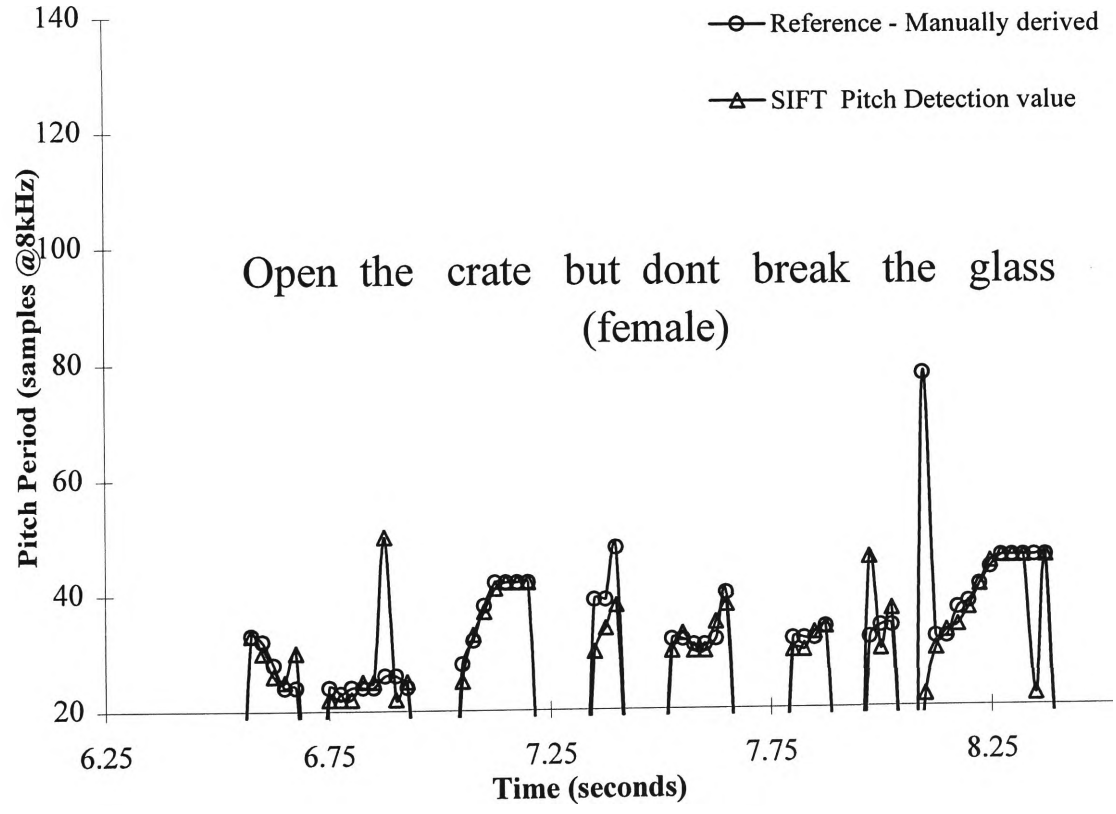

Figure A4 - SIFT Pitch Detector generated Pitch Profile

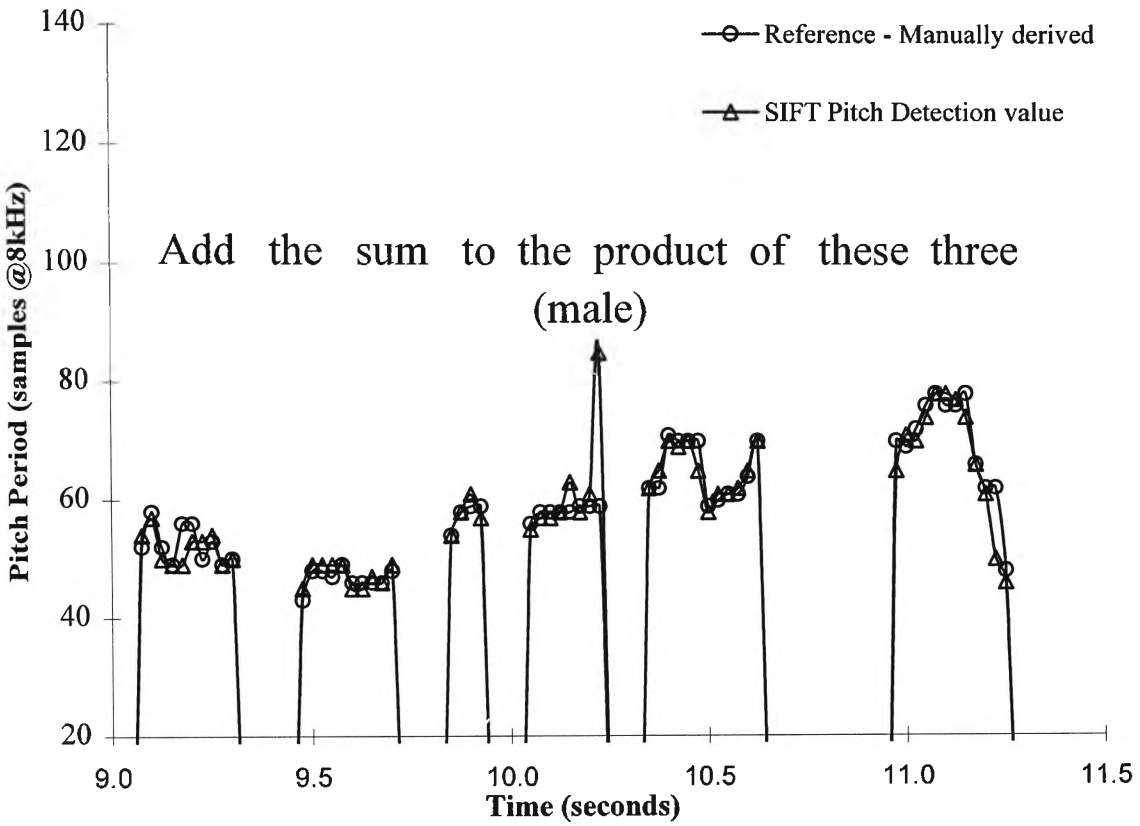

Figure A5 - SIFT Pitch Detector generated Pitch Profile

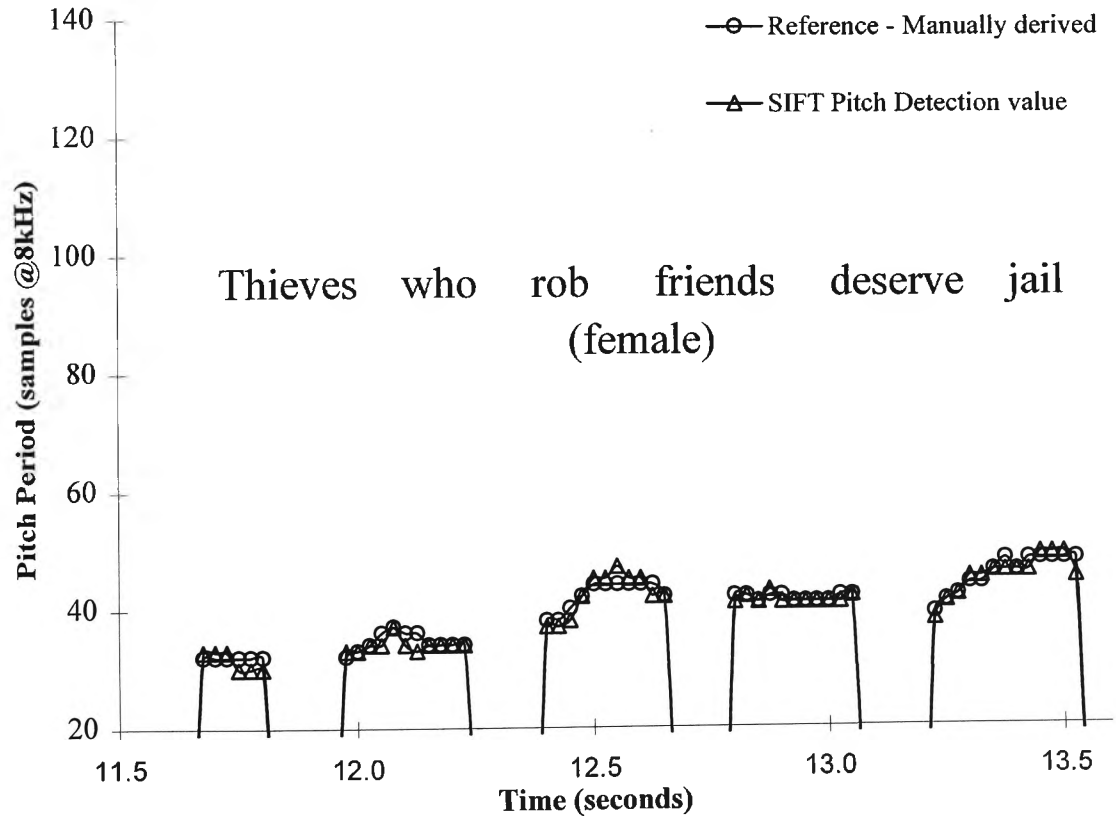

Figure A6 - SIFT Pitch Detector generated Pitch Profile

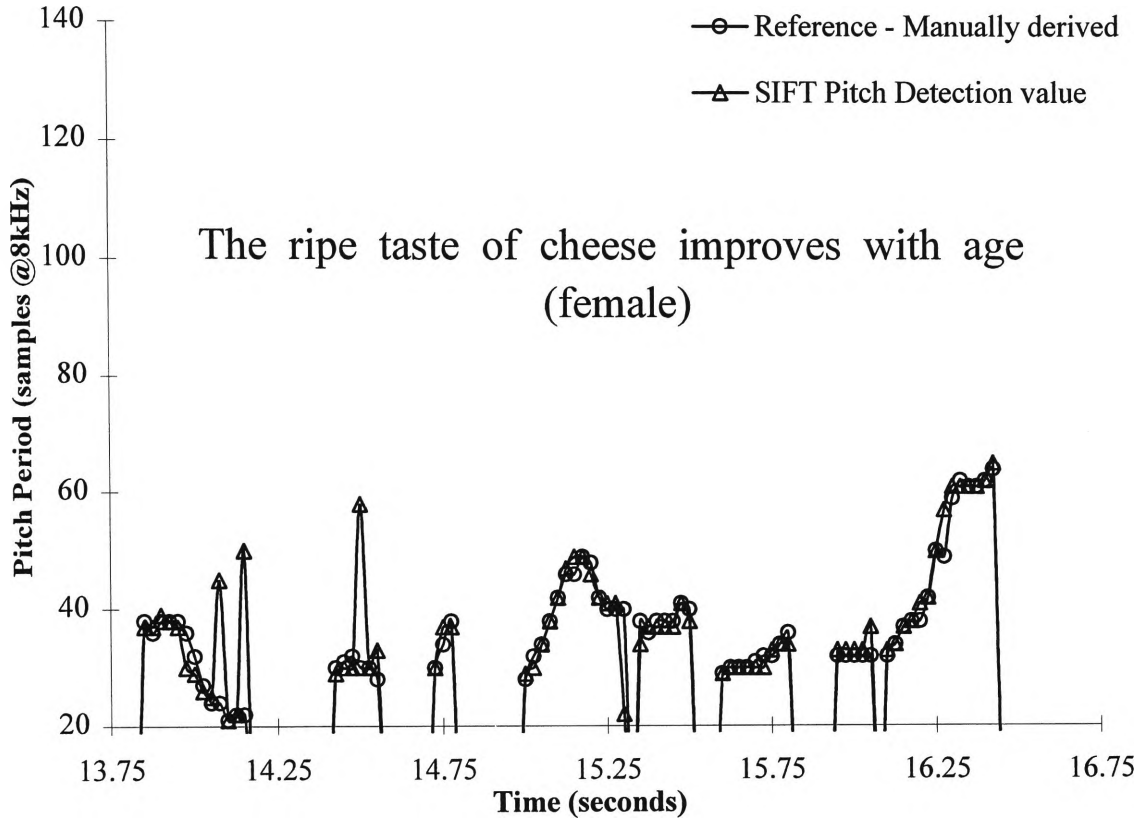

Figure A7 - SIFT Pitch Detector generated Pitch Profile

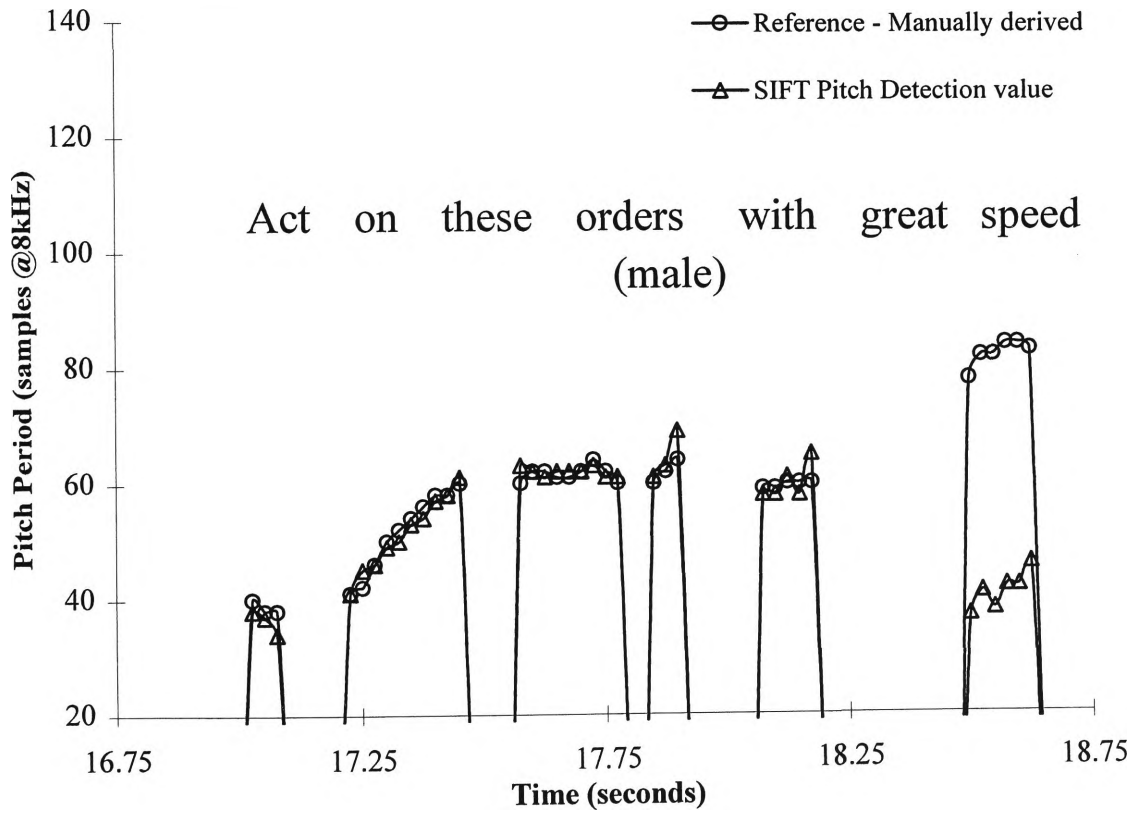

Figure A8 - SIFT Pitch Detector generated Pitch Profile

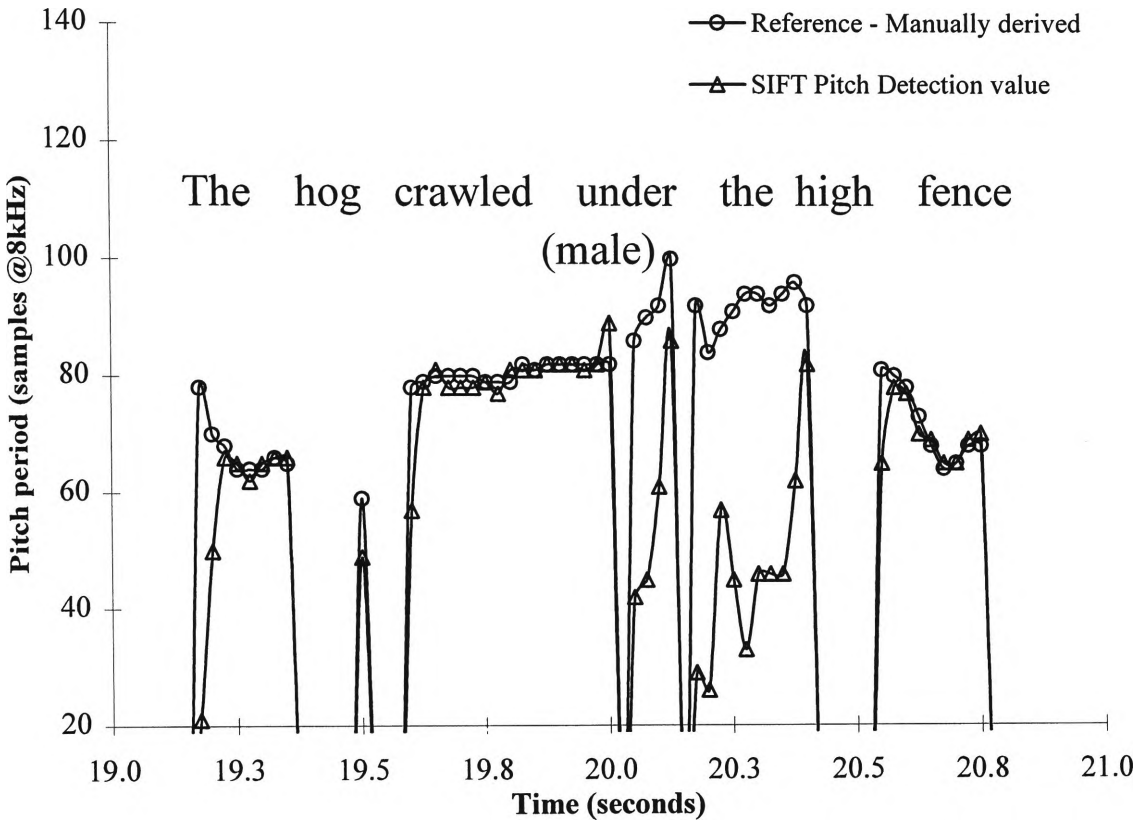

Figure A9 - SIFT Pitch Detector generated Pitch Profile

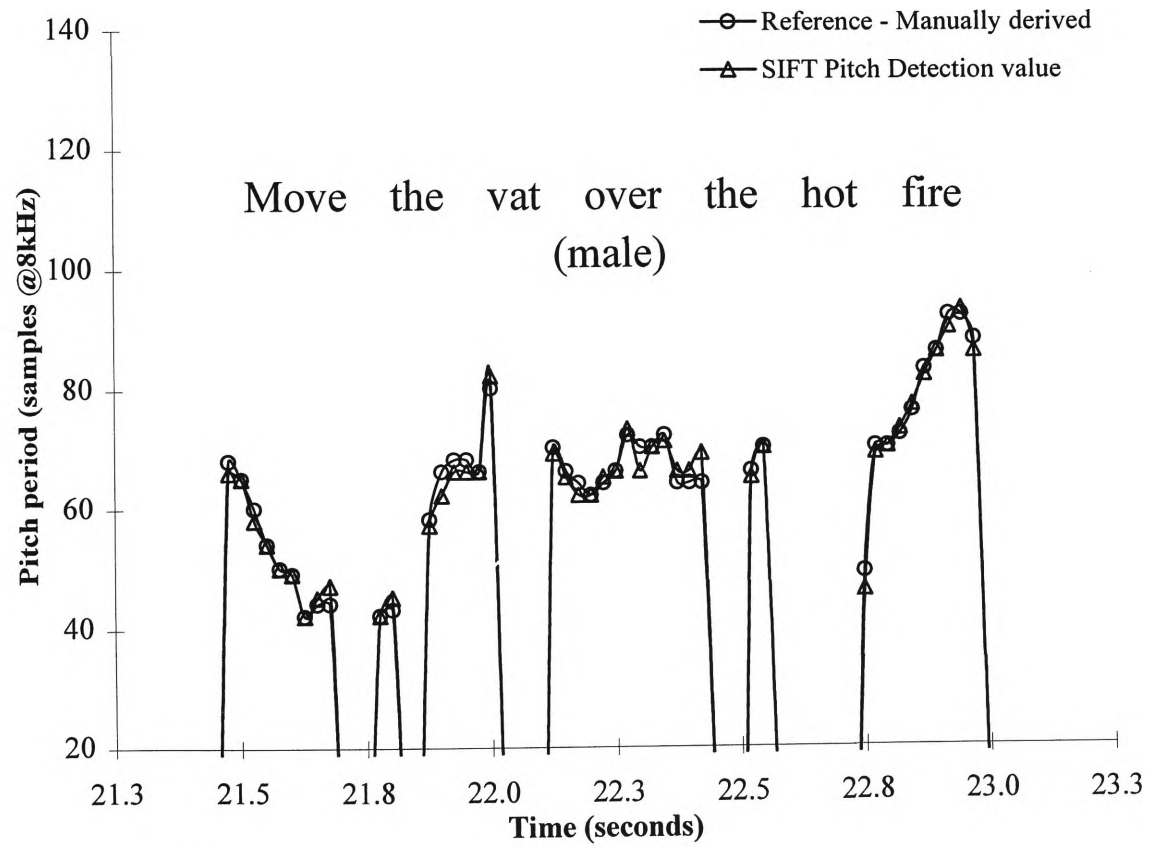

Figure A10 - SIFT Pitch Detector generated Pitch Profile

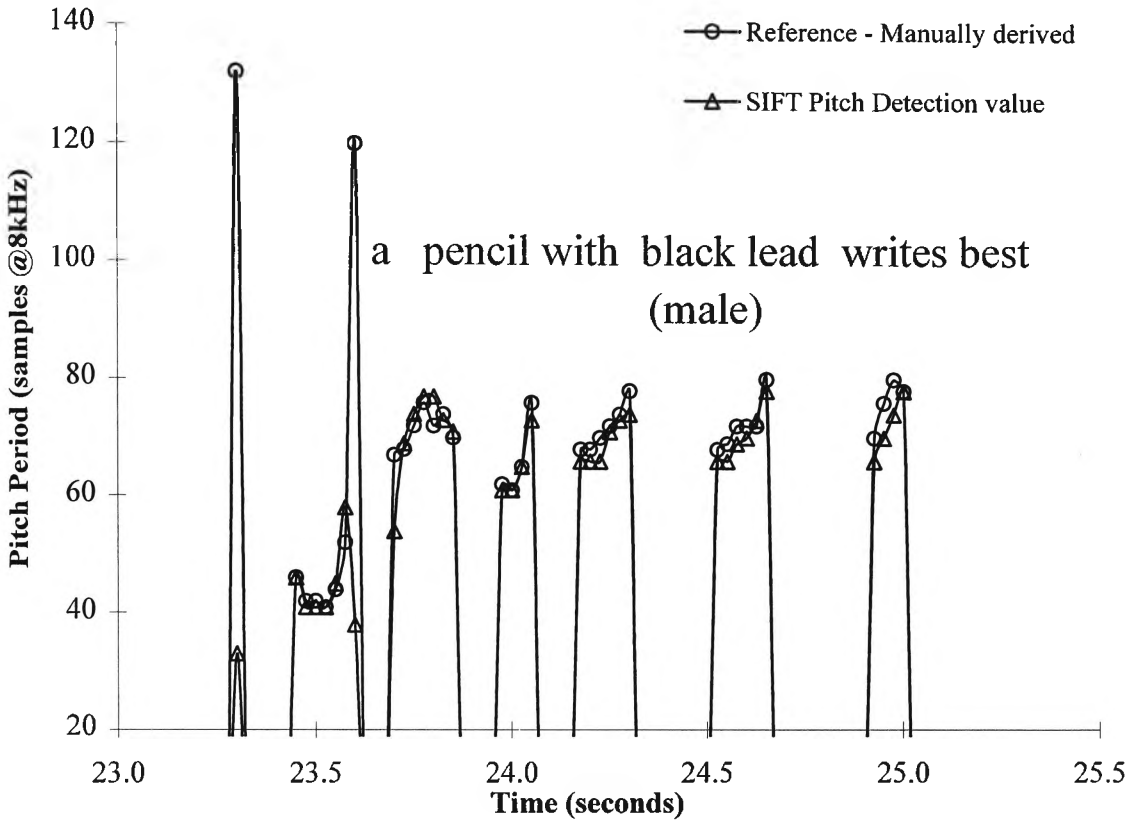

Figure A11 - SIFT Pitch Detector generated Pitch Profile

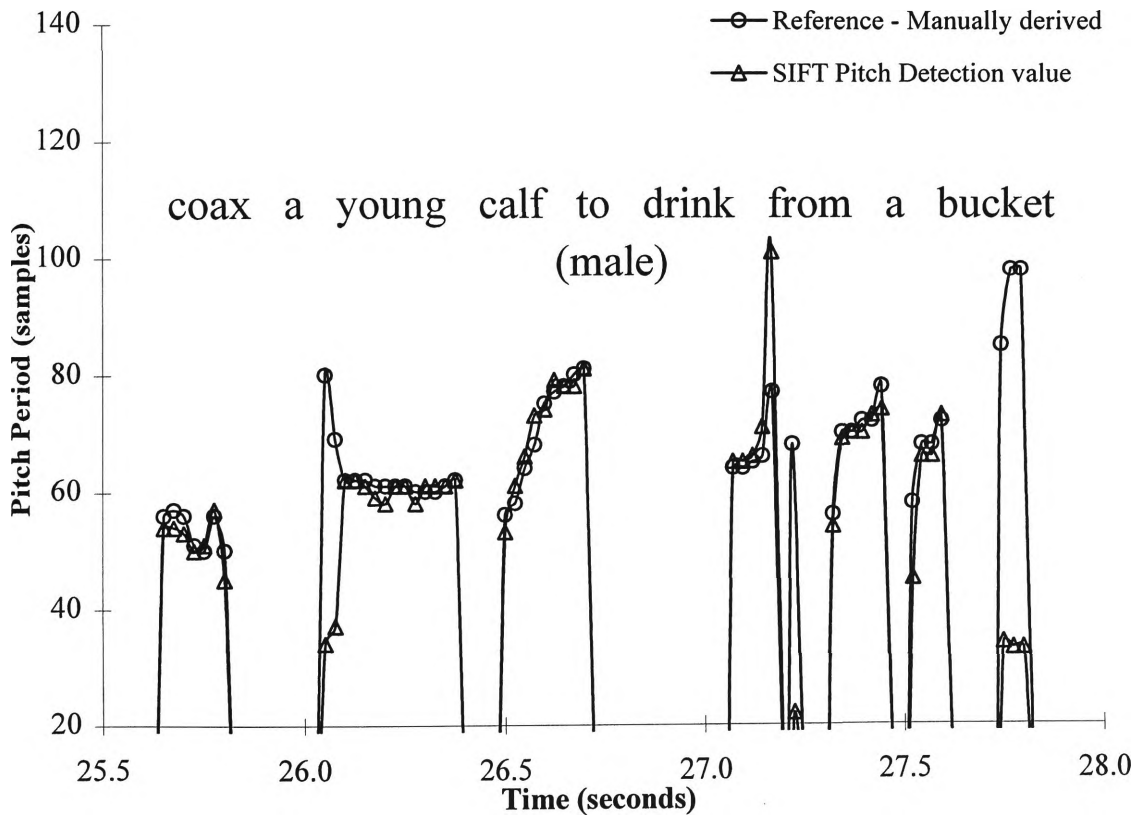

Figure A12 - SIFT Pitch Detector generated Pitch Profile

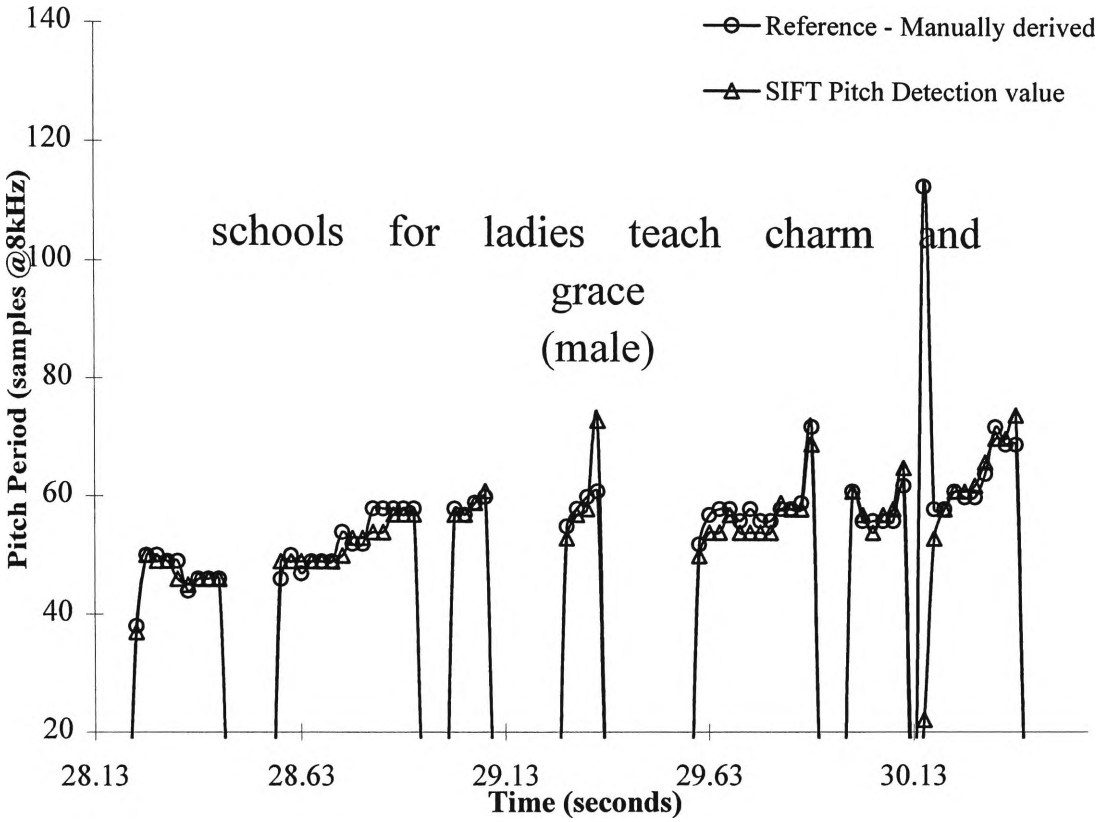

Figure A13 - SIFT Pitch Detector generated Pitch Profile

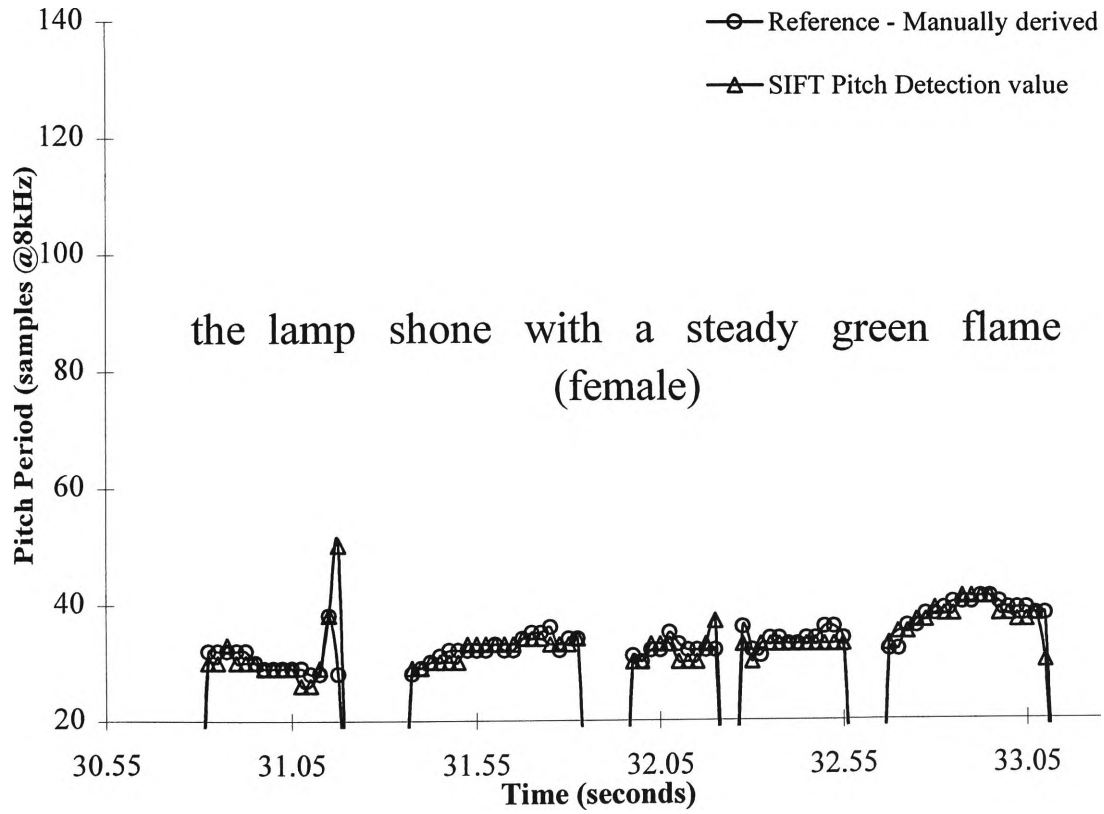

Figure A14 - SIFT Pitch Detector generated Pitch Profile

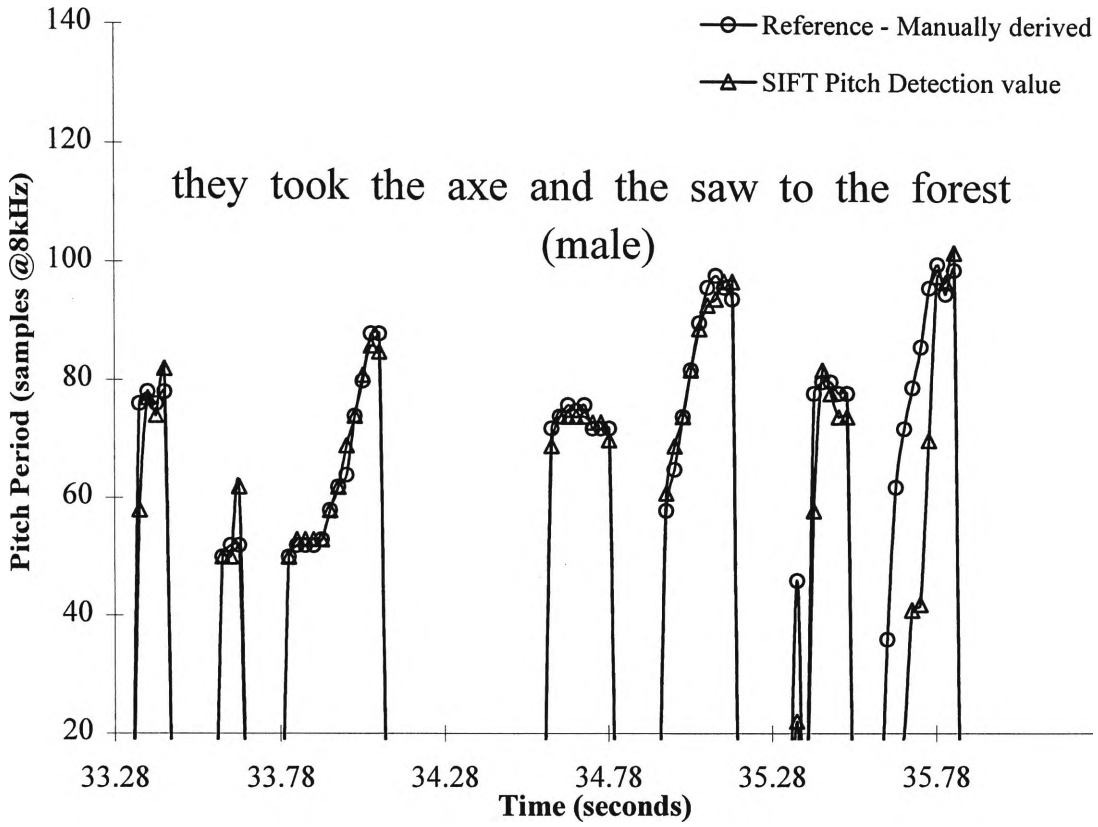

Figure A15 - SIFT Pitch Detector generated Pitch Profile

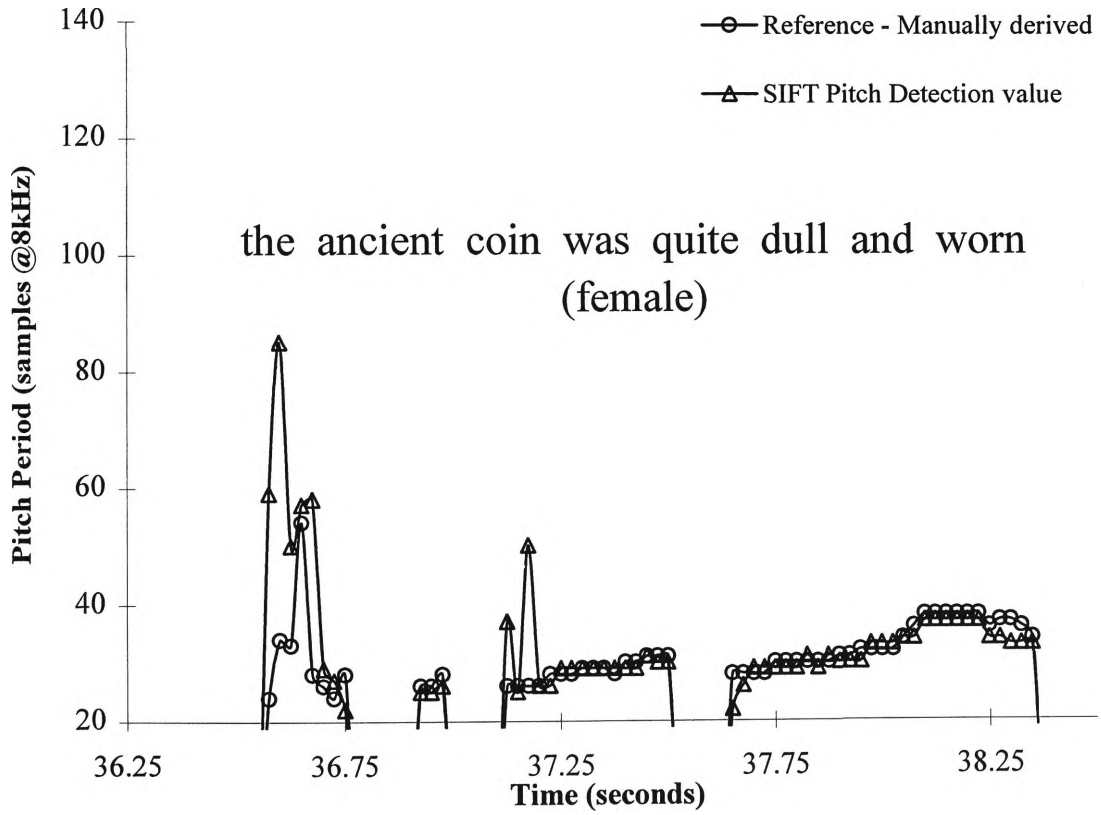

Figure A16 - SIFT Pitch Detector generated Pitch Profile

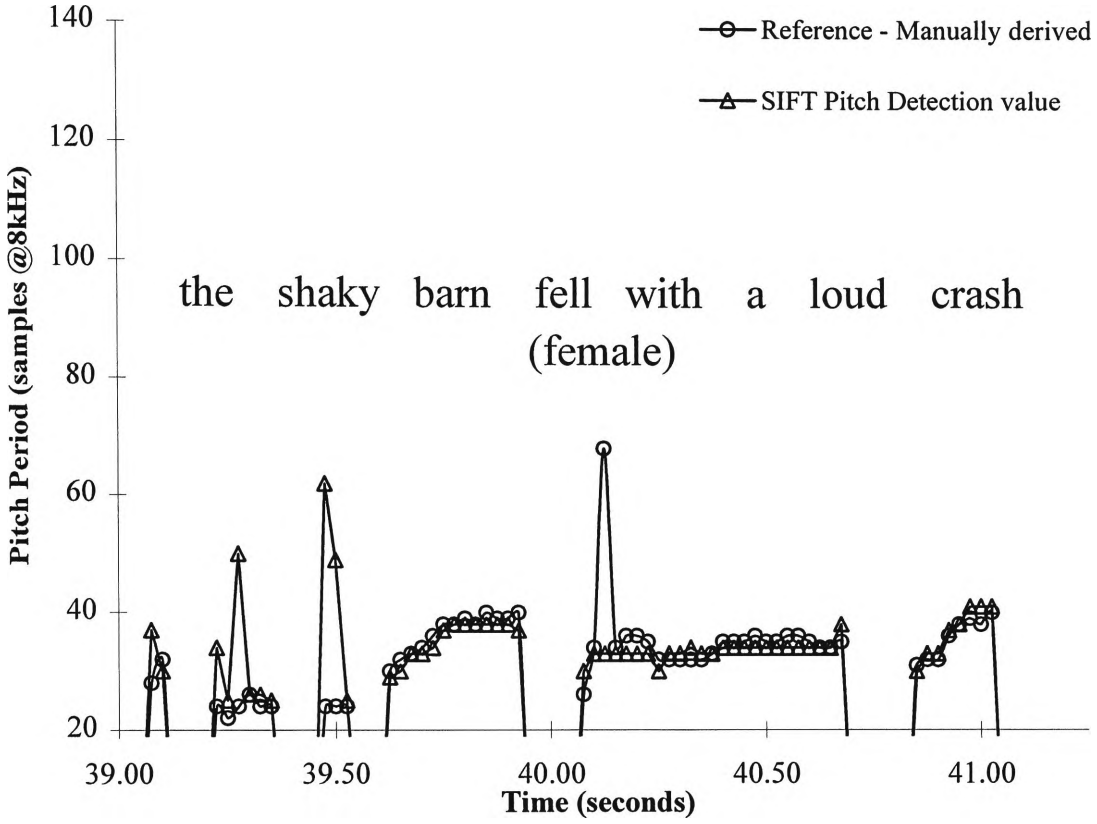

Figure A17 - SIFT Pitch Detector generated Pitch Profile

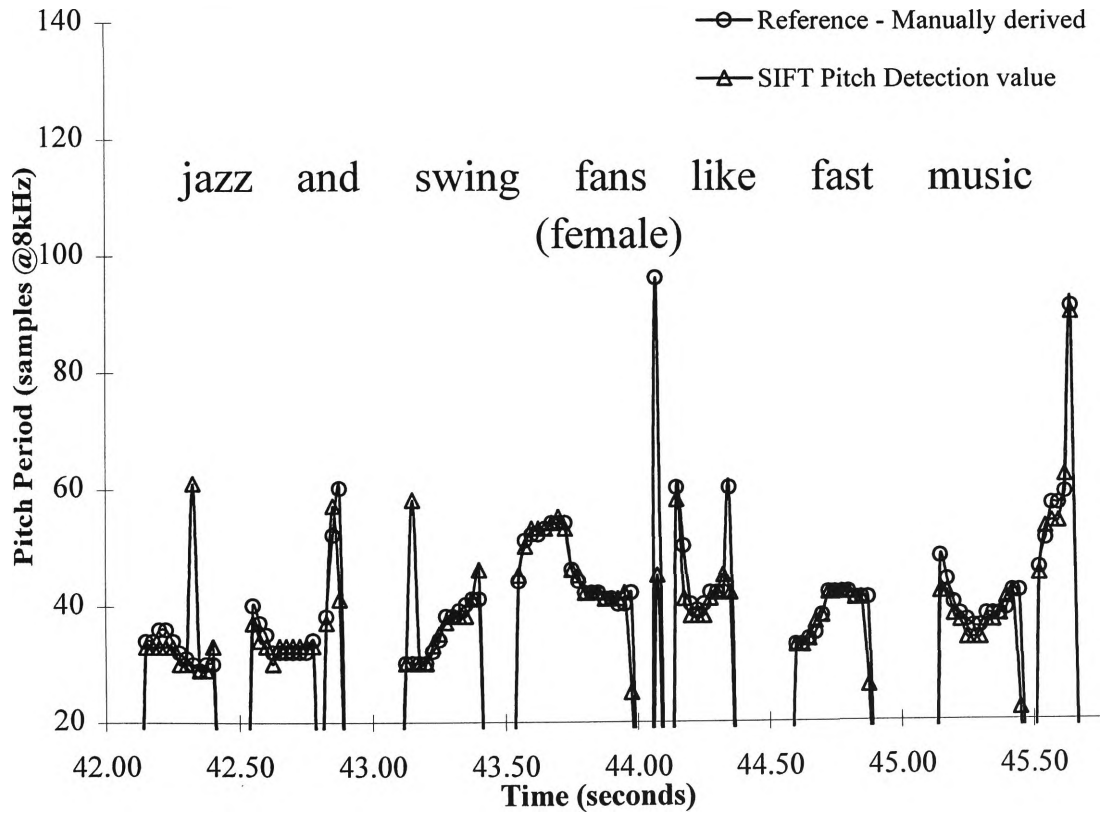

Figure A18 - SIFT Pitch Detector generated Pitch Profile

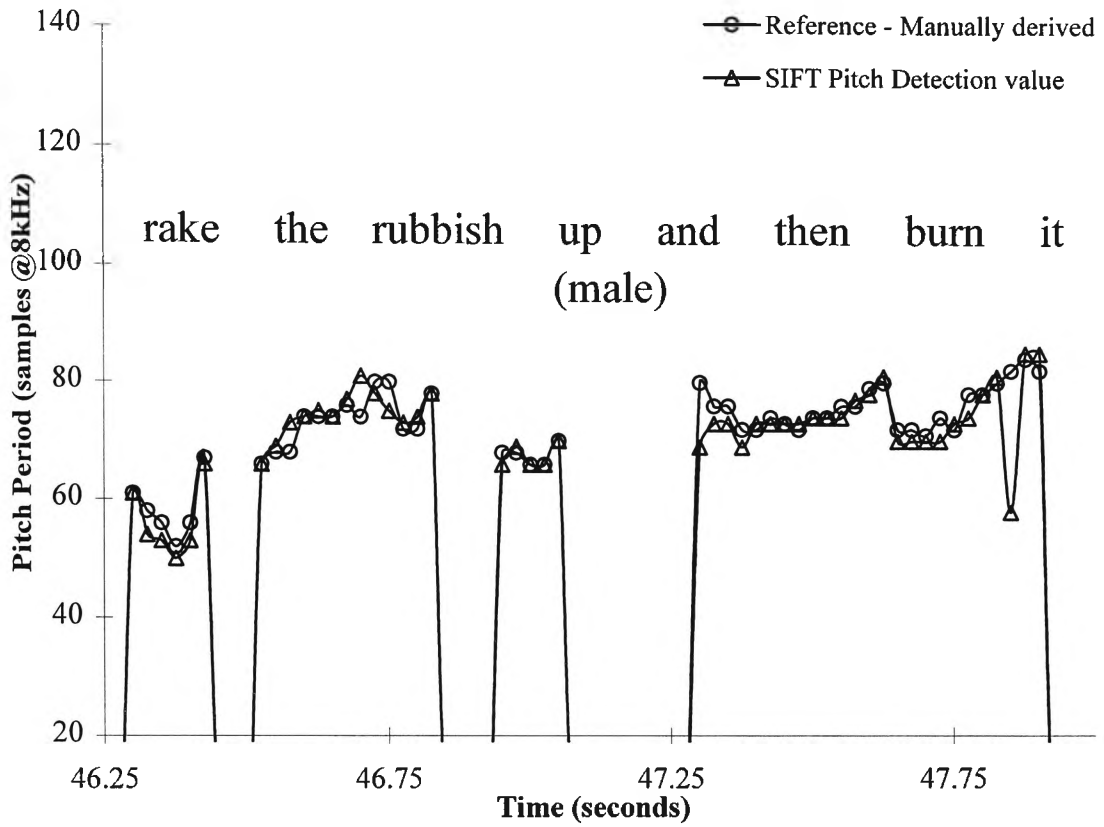

Figure A19 - SIFT Pitch Detector generated Pitch Profile

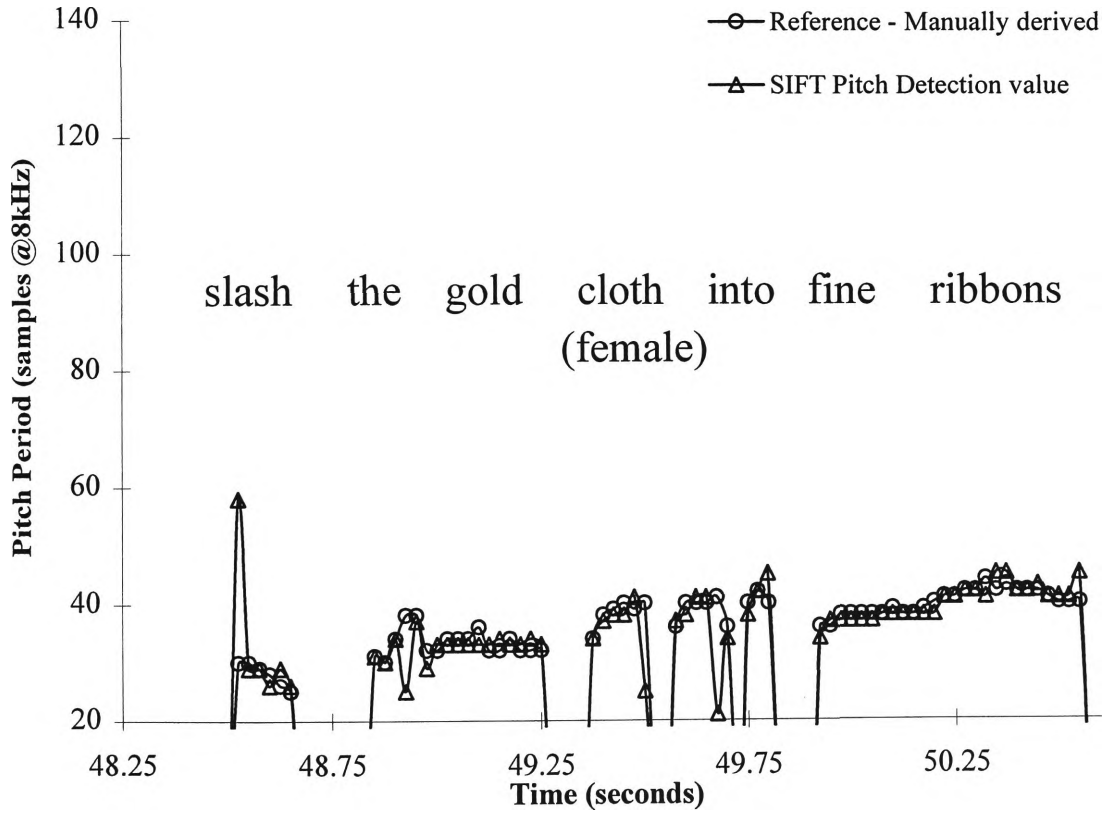

Figure A20 - SIFT Pitch Detector generated Pitch Profile

APPENDIX B

Generated Pitch Profiles

using

Glottal Closure Interval

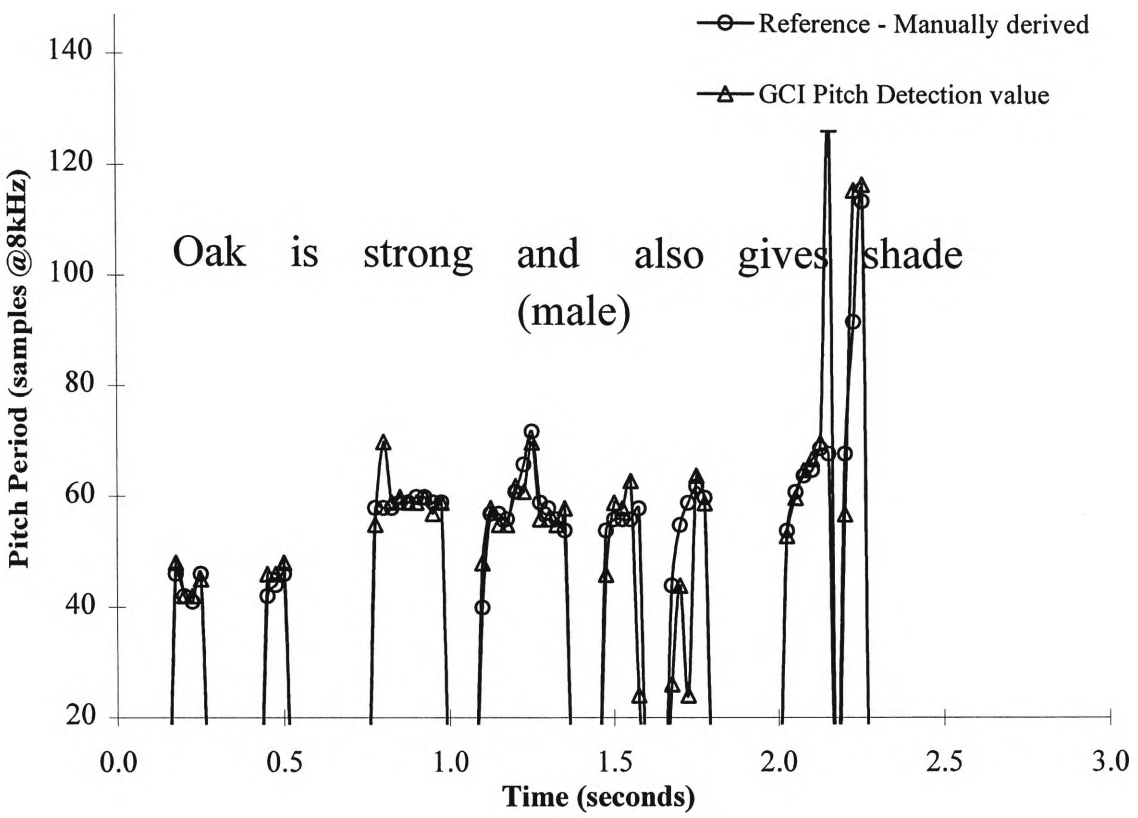

Figure B1 - GCI Pitch Detector generated Pitch Profile

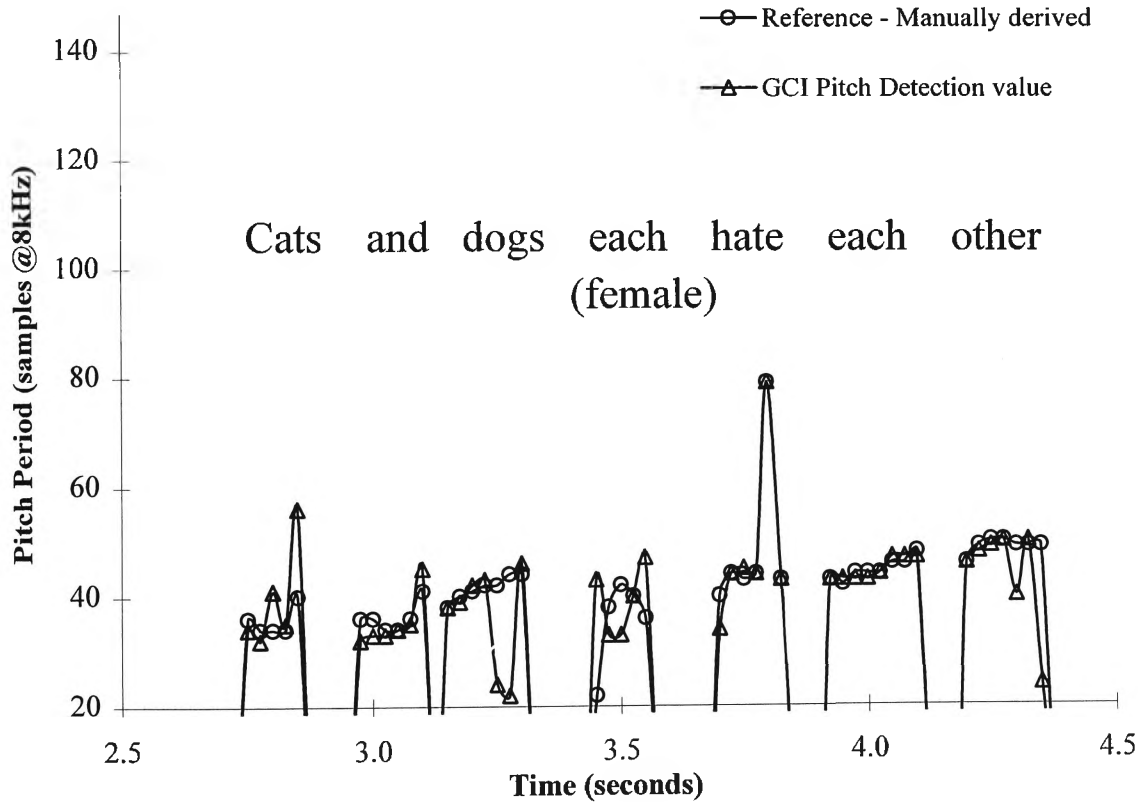

Figure B2 - GCI Pitch Detector generated Pitch Profile

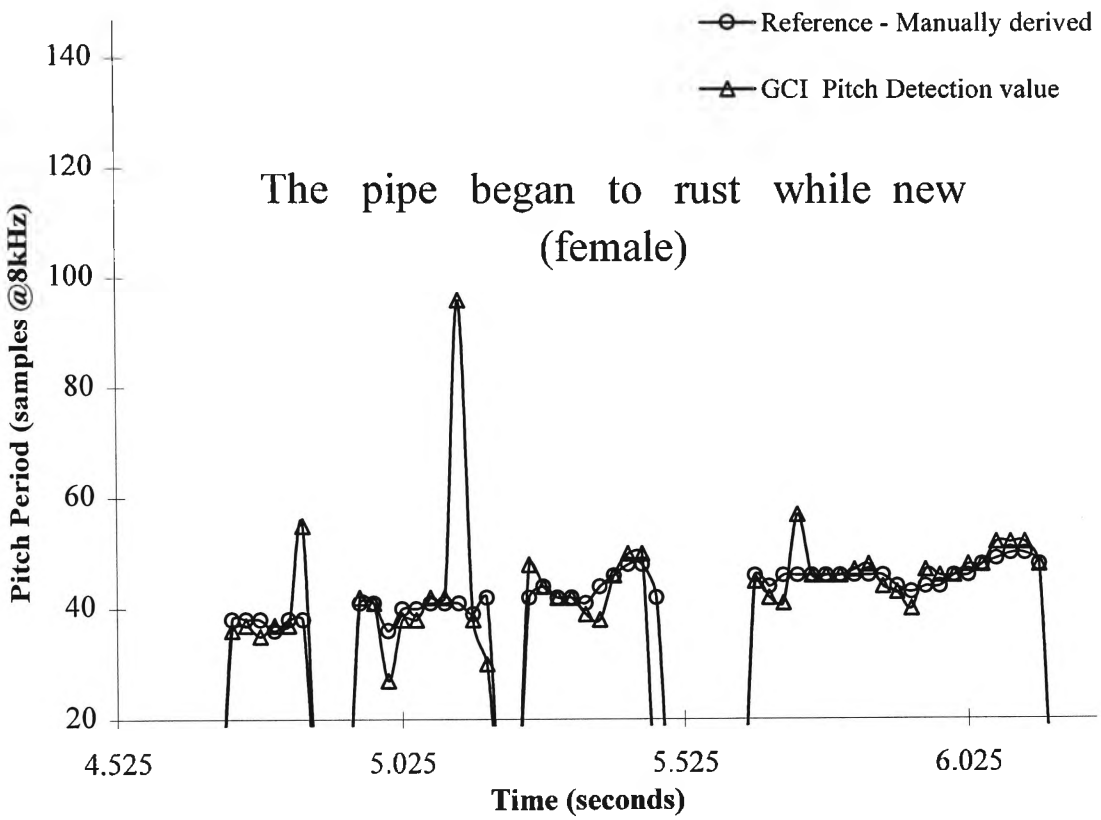

Figure B3 - GCI Pitch Detector generated Pitch Profile

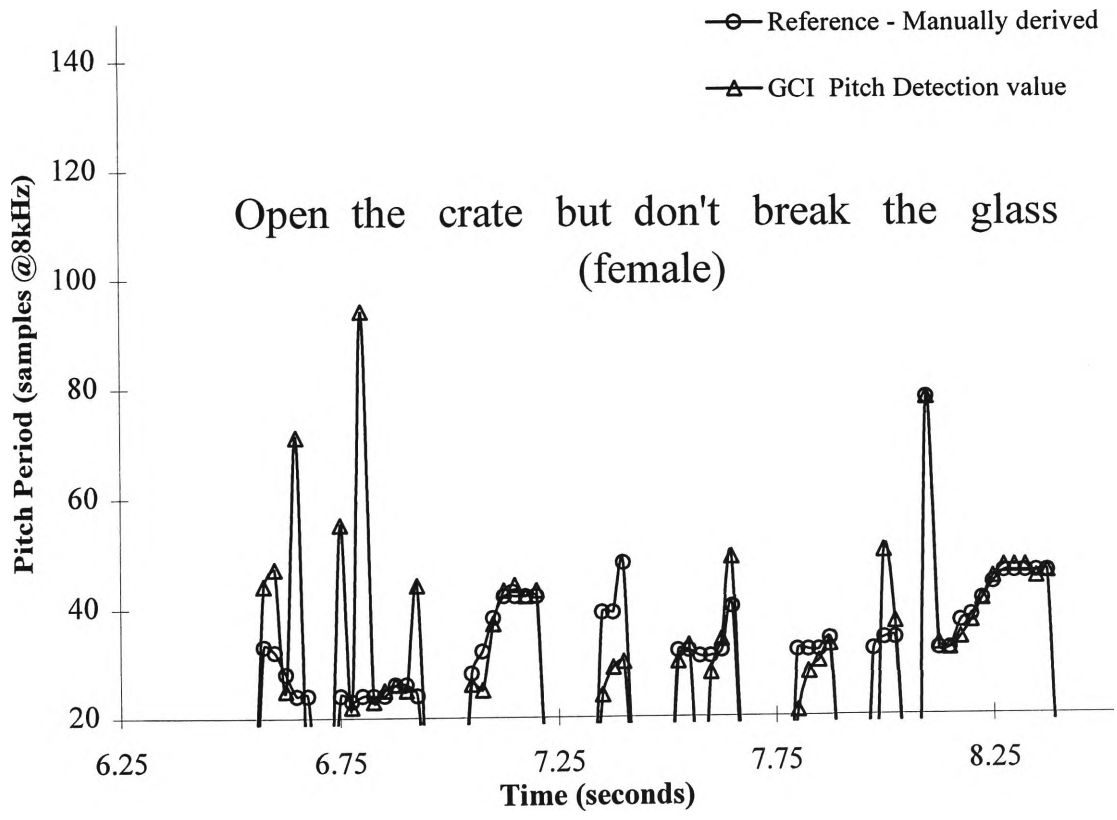

Figure B4 - GCI Pitch Detector generated Pitch Profile

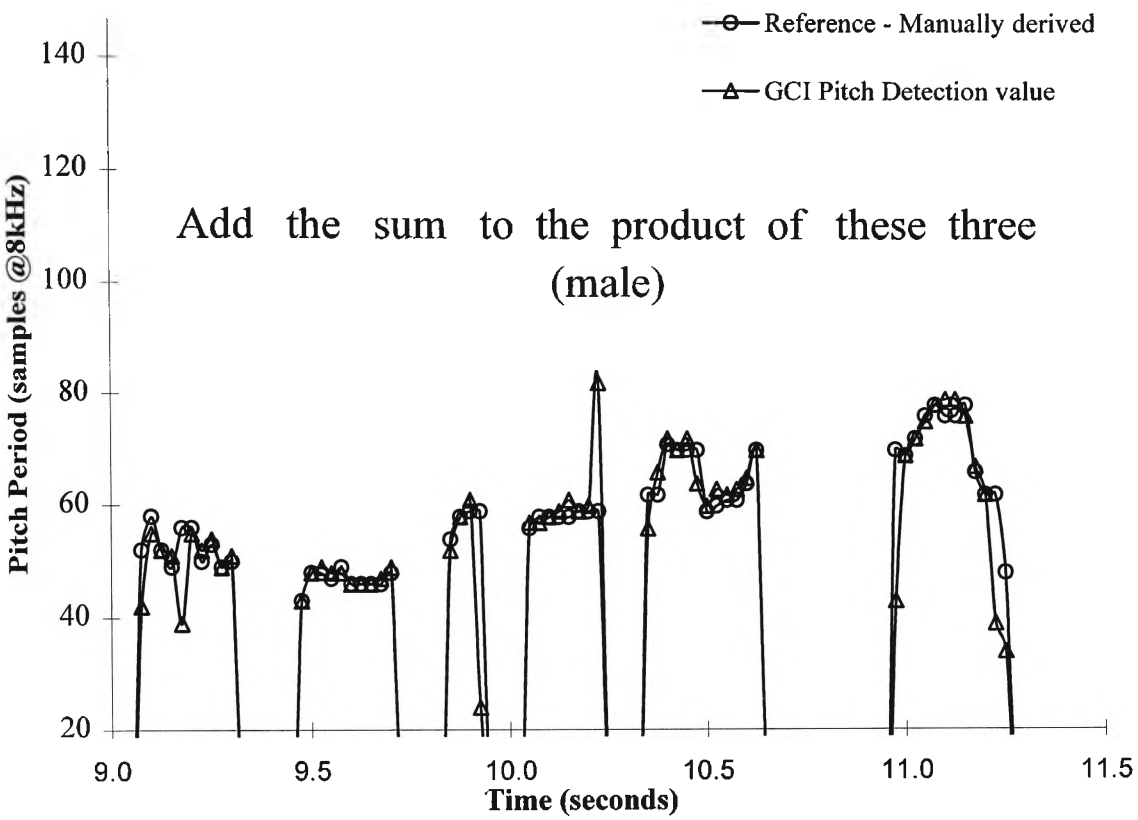

Figure B5 - GCI Pitch Detector generated Pitch Profile

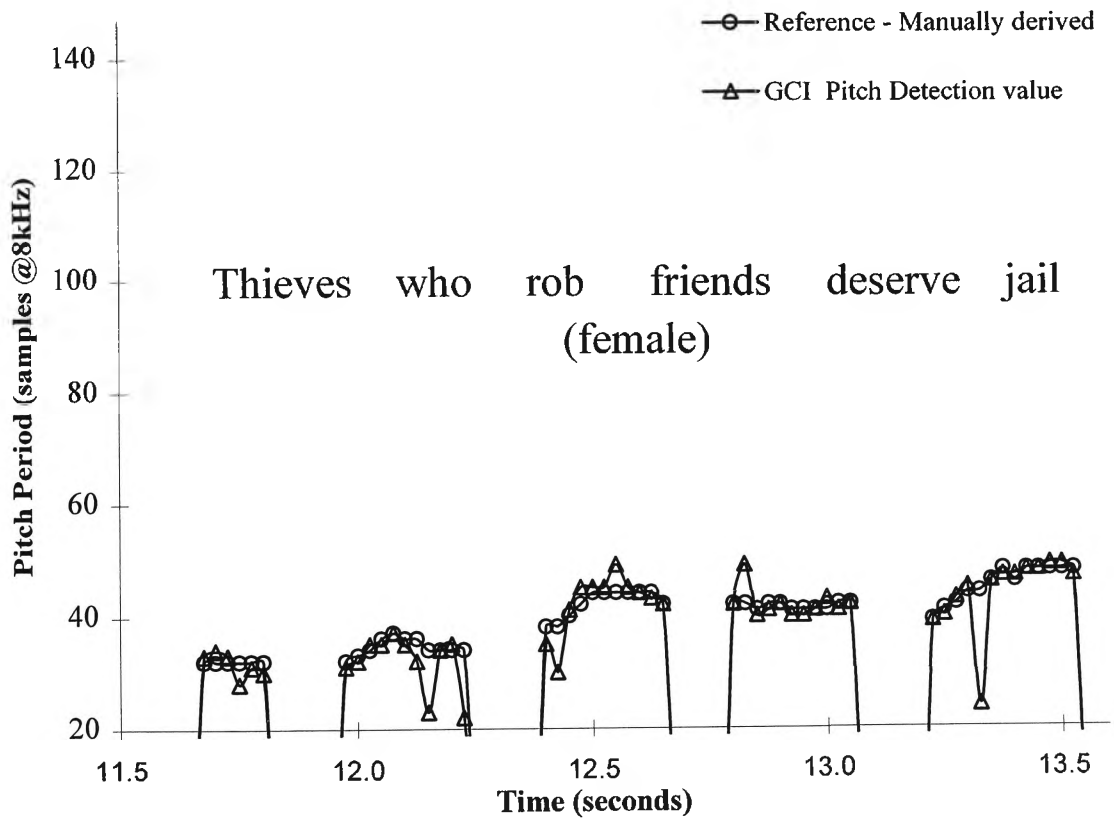

Figure B6 - GCI Pitch Detector generated Pitch Profile

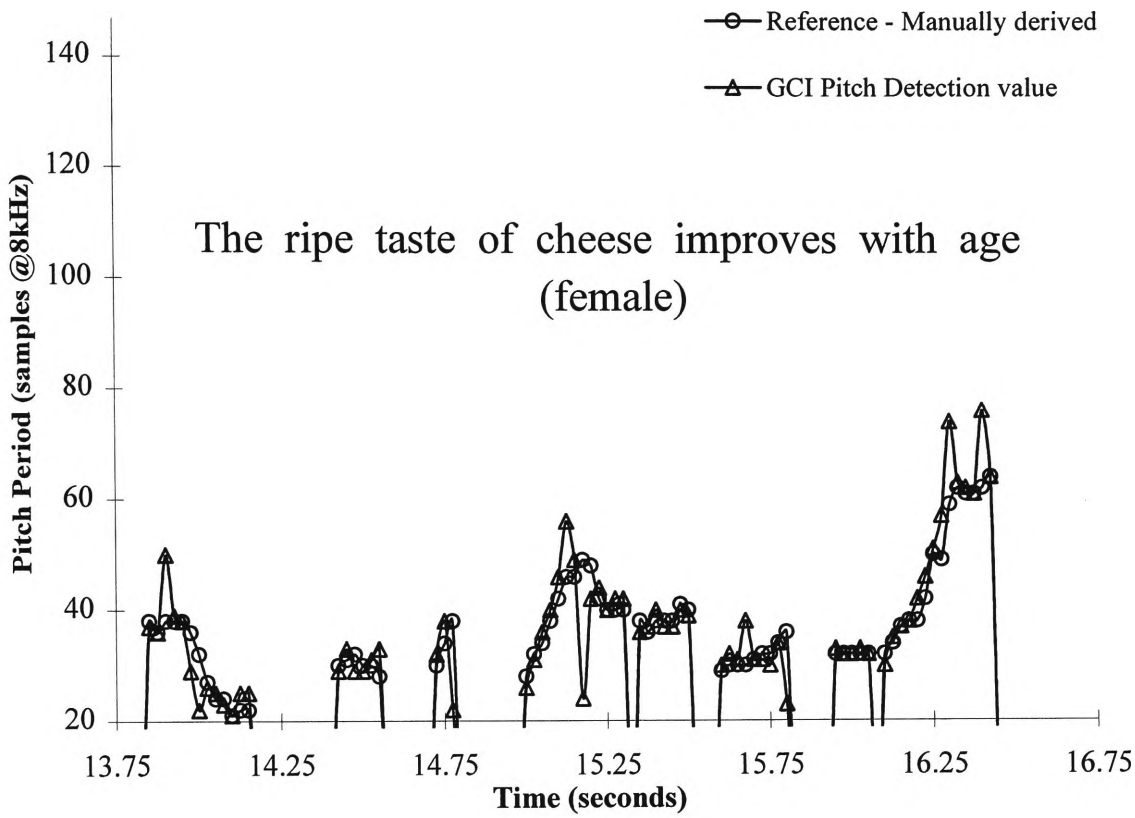

Figure B7 - GCI Pitch Detector generated Pitch Profile

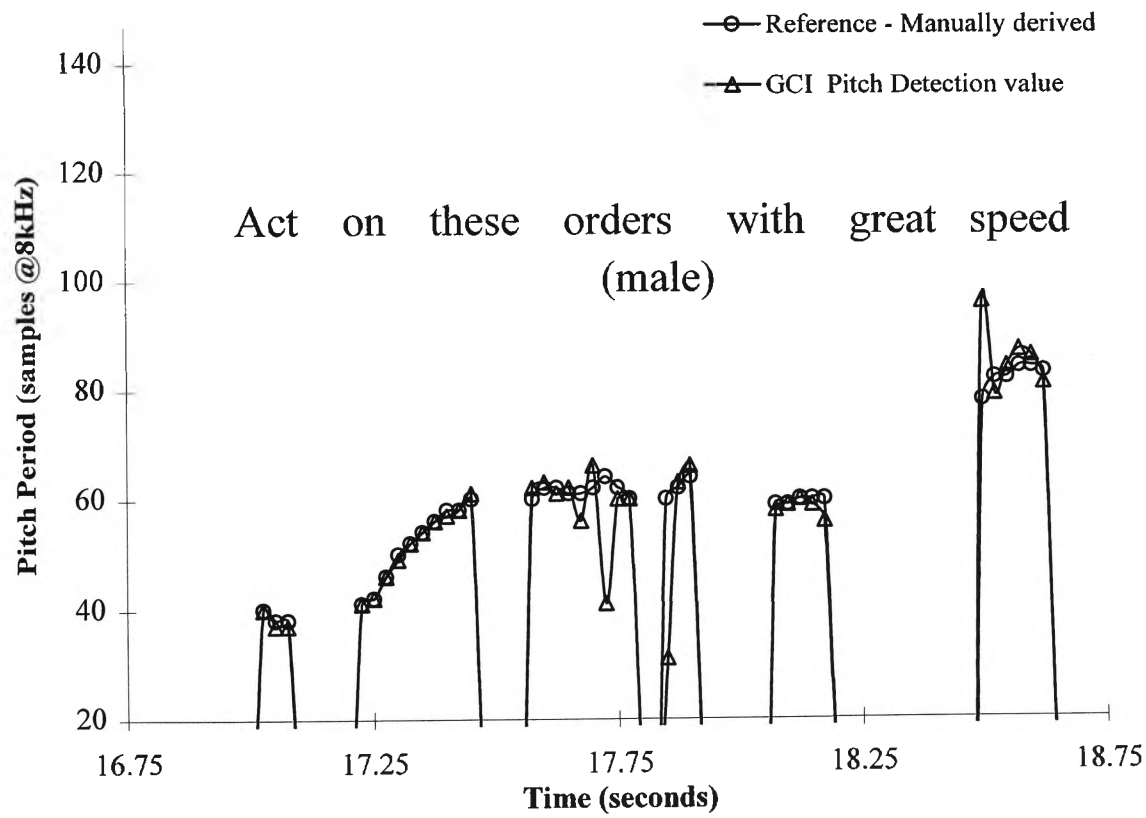

Figure B8 - GCI Pitch Detector generated Pitch Profile

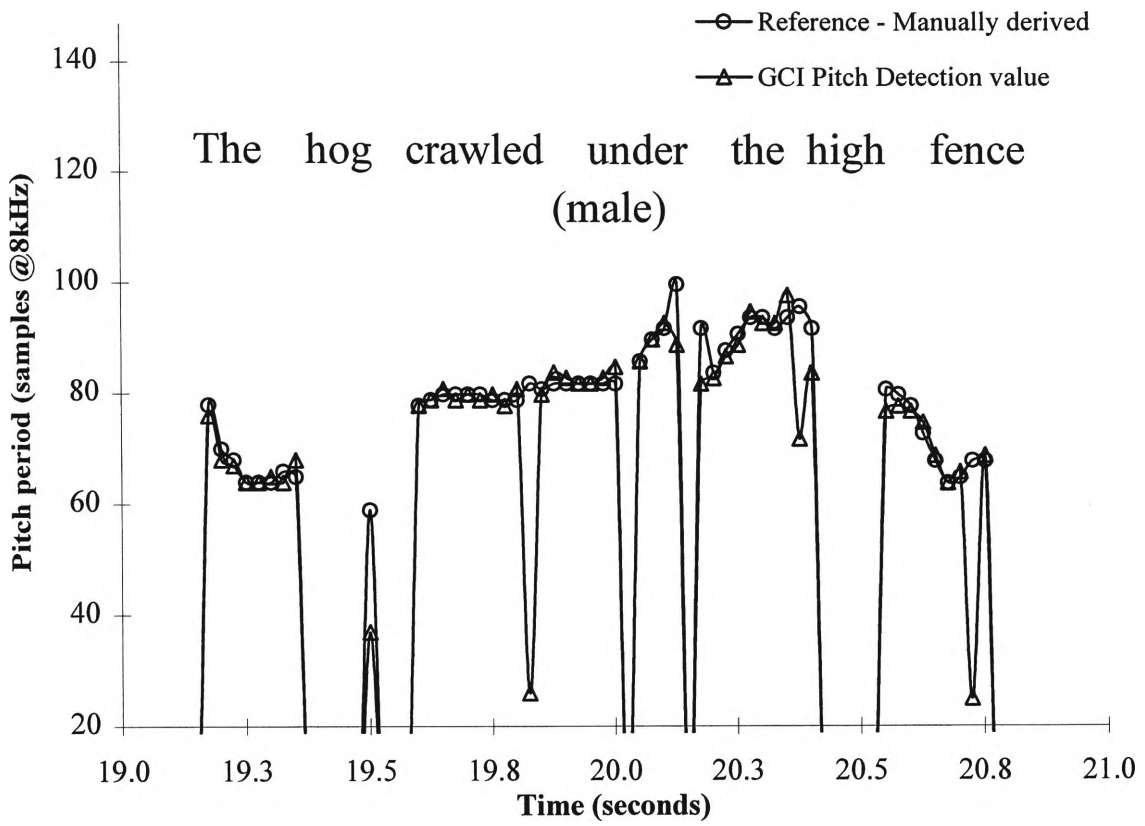

Figure B9 - GCI Pitch Detector generated Pitch Profile

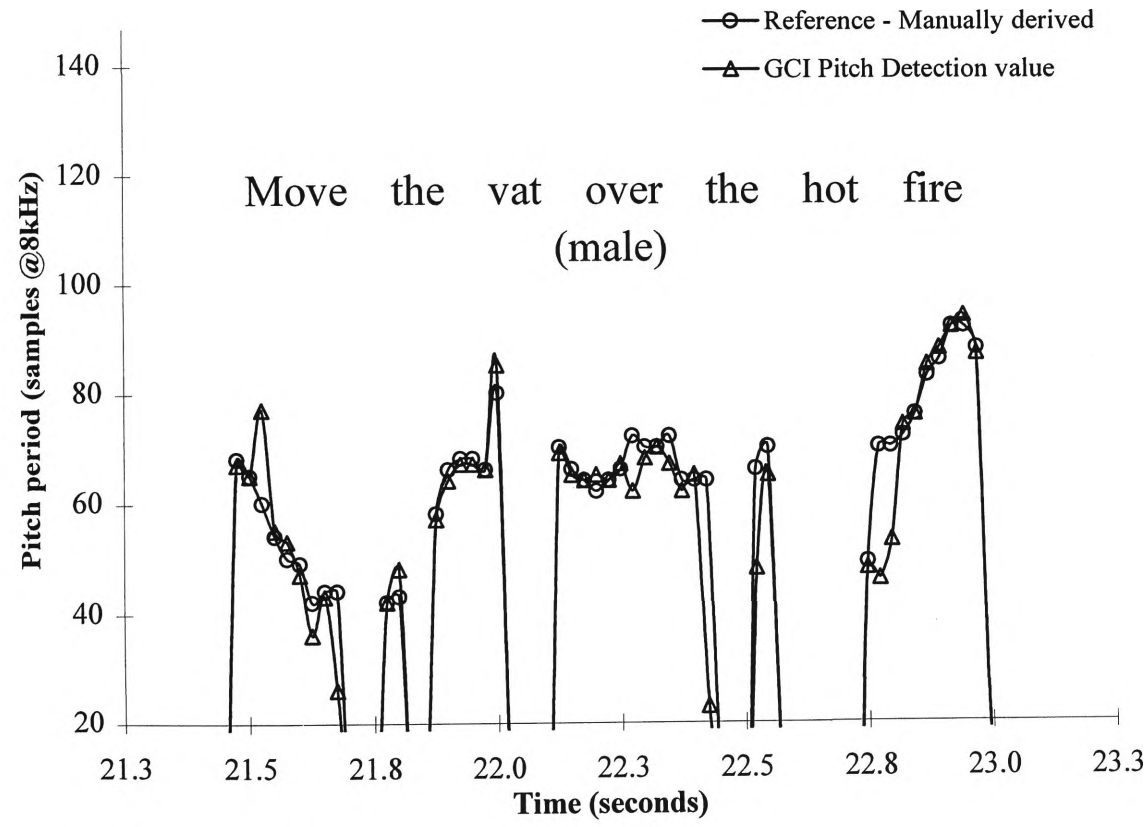

Figure B10 - GCI Pitch Detector generated Pitch Profile

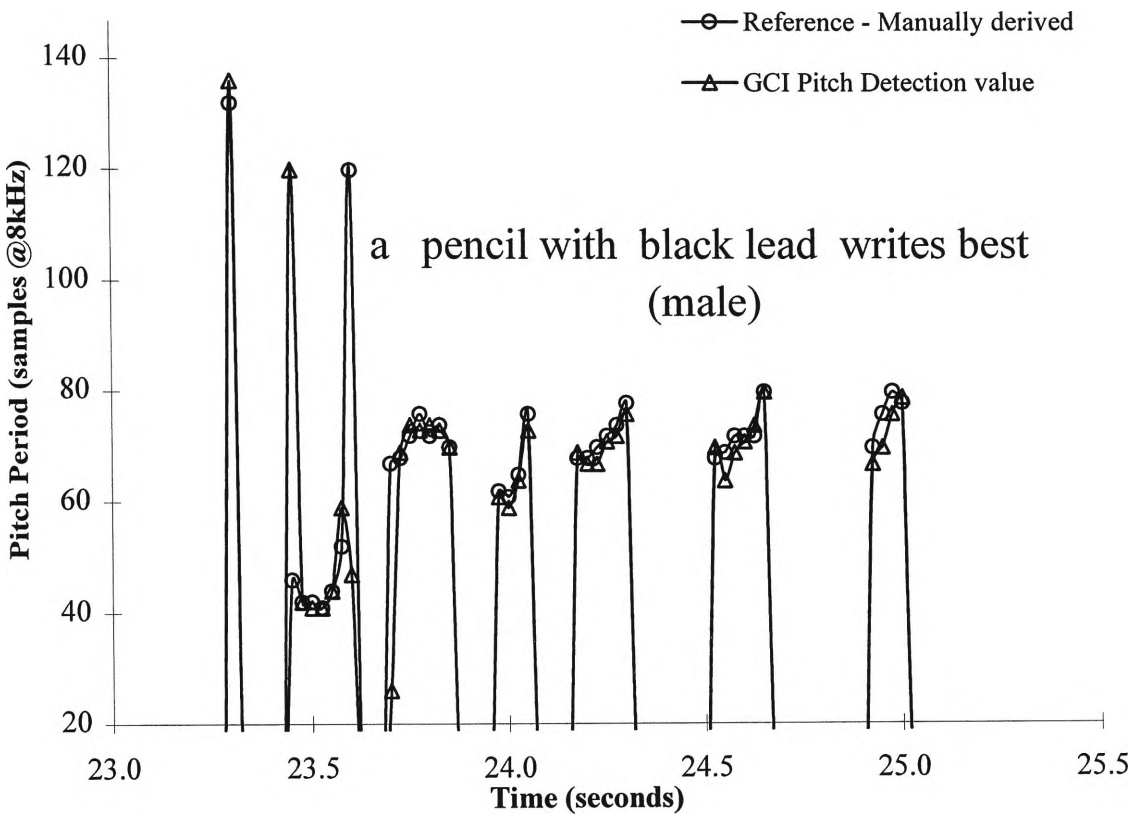

Figure B11 - GCI Pitch Detector generated Pitch Profile

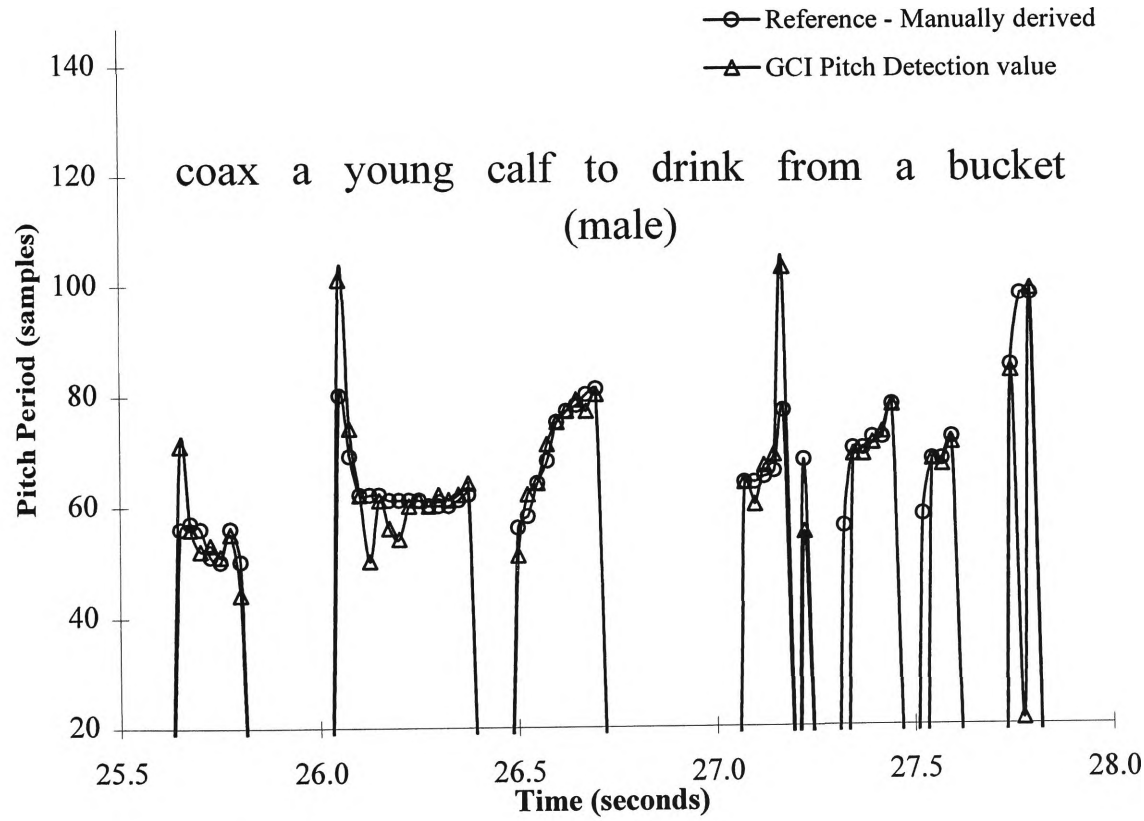

Figure B12 - GCI Pitch Detector generated Pitch Profile

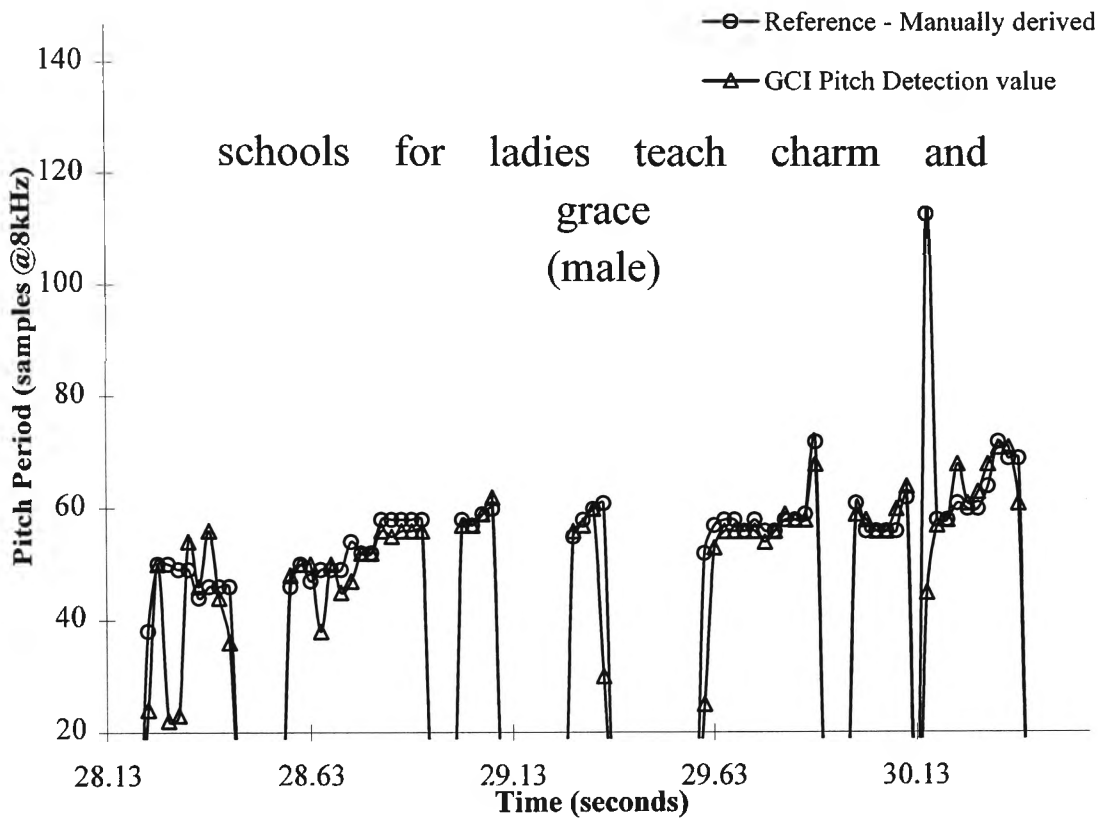

Figure B13 - GCI Pitch Detector generated Pitch Profile

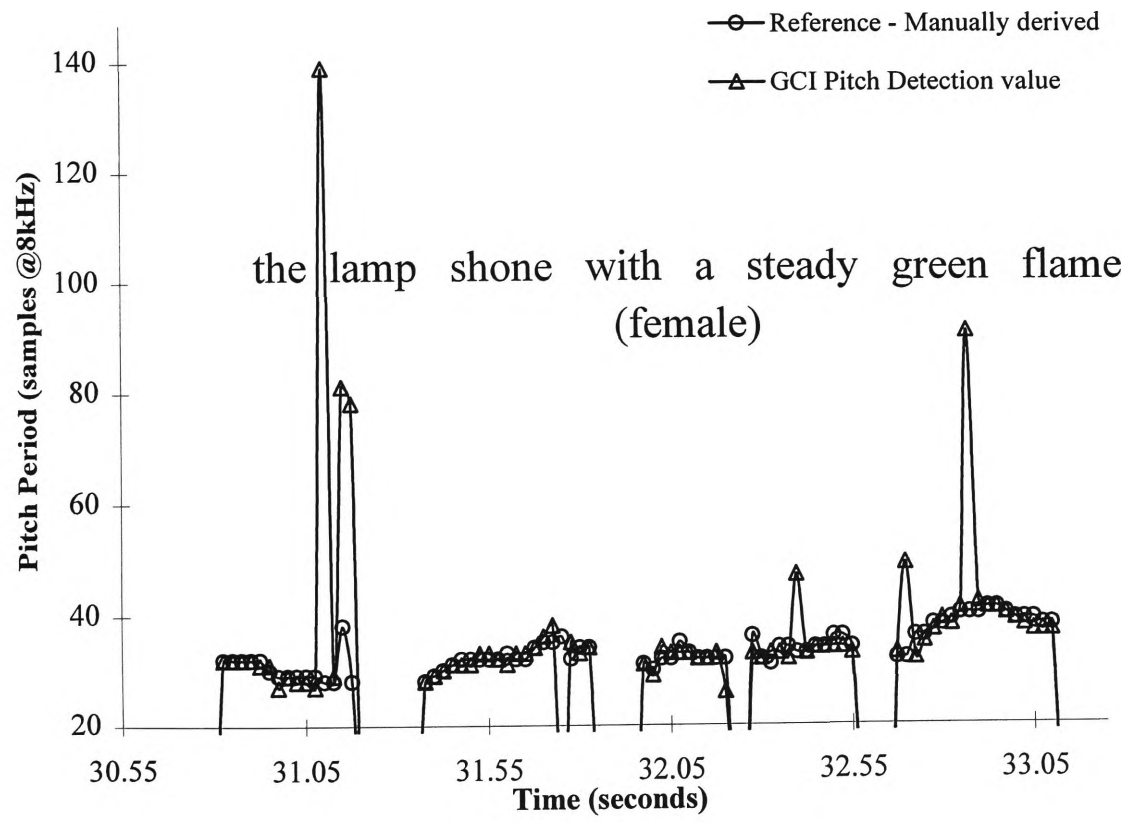

Figure B14 - GCI Pitch Detector generated Pitch Profile

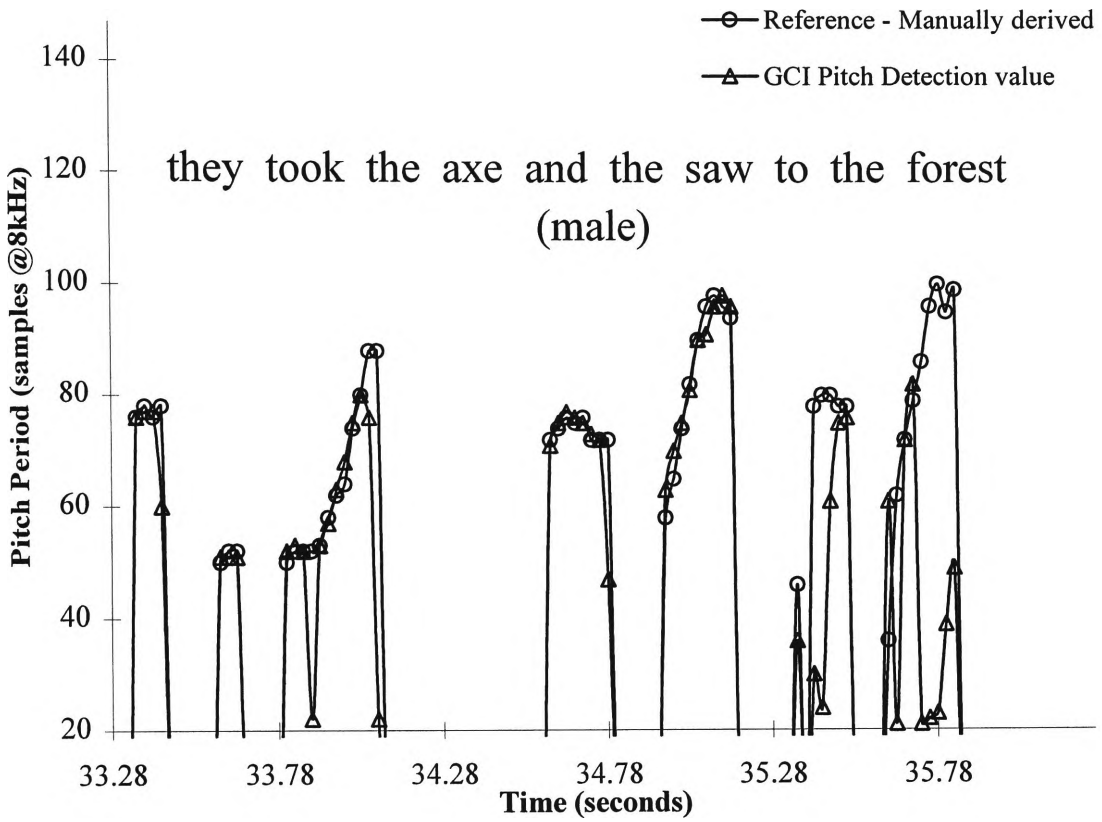

Figure B15 - GCI Pitch Detector generated Pitch Profile

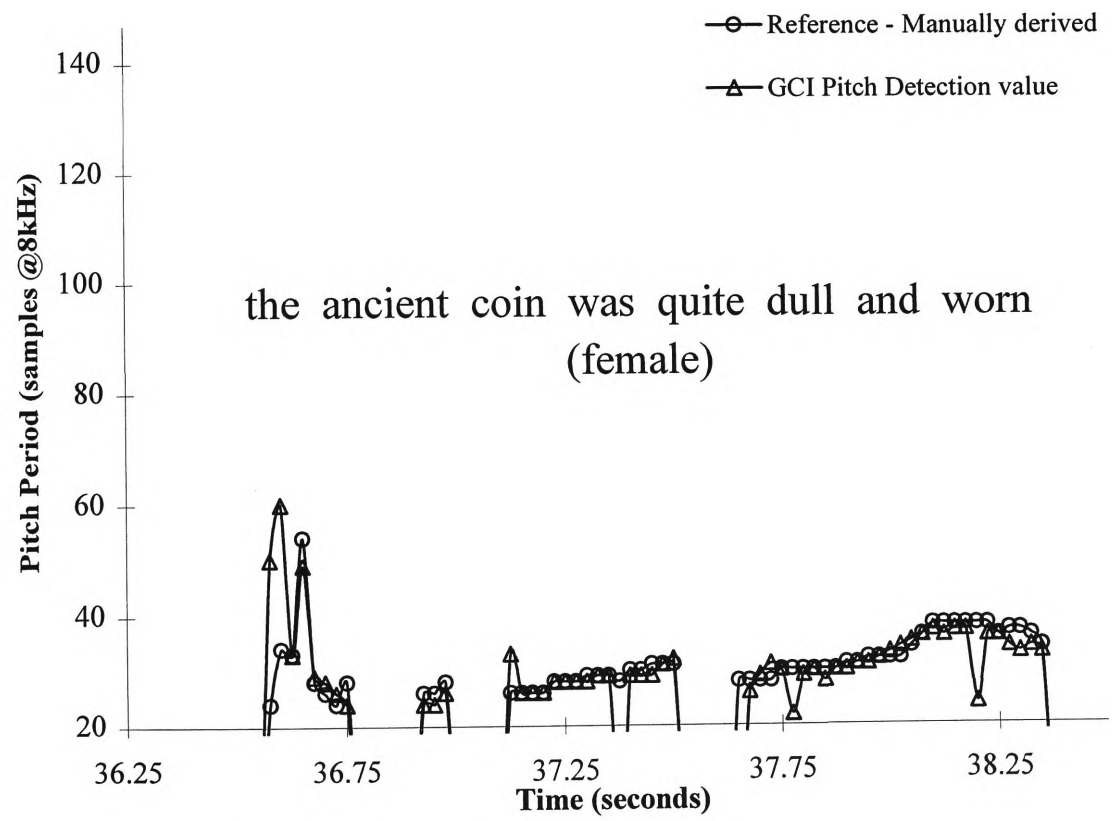

Figure B16 - GCI Pitch Detector generated Pitch Profile

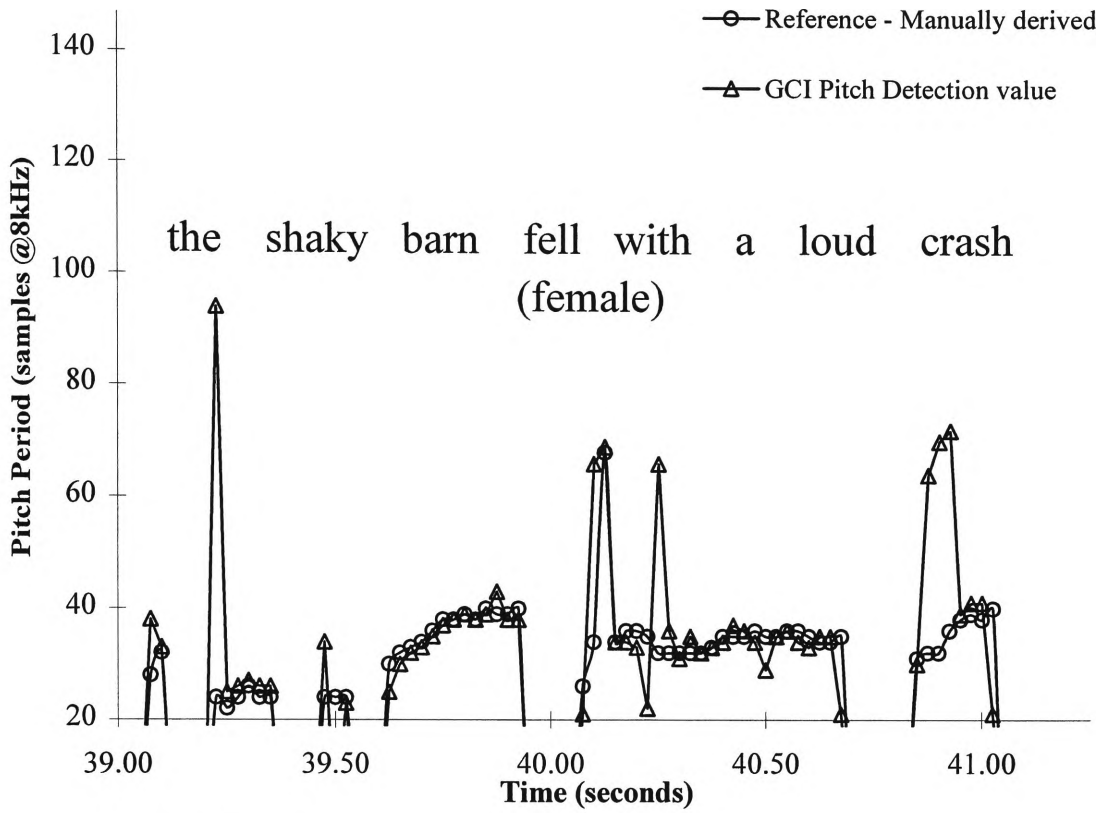

Figure B17 - GCI Pitch Detector generated Pitch Profile

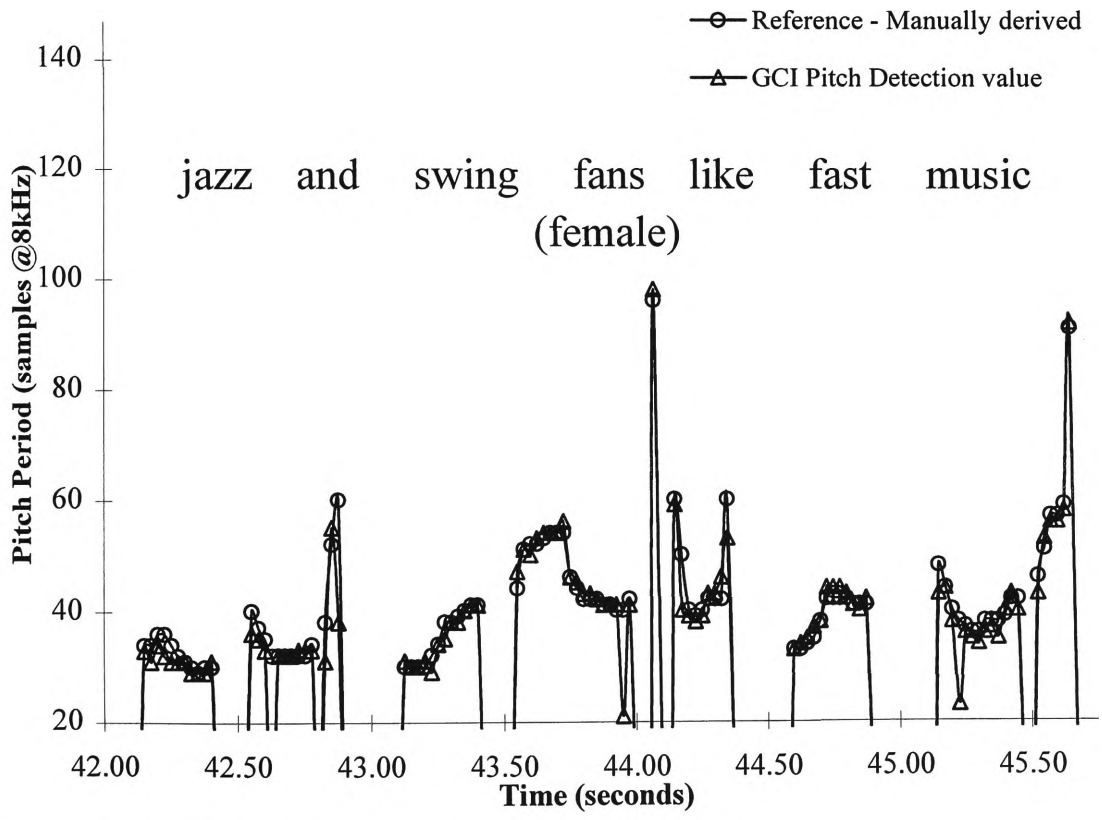

Figure B18 - GCI Pitch Detector generated Pitch Profile

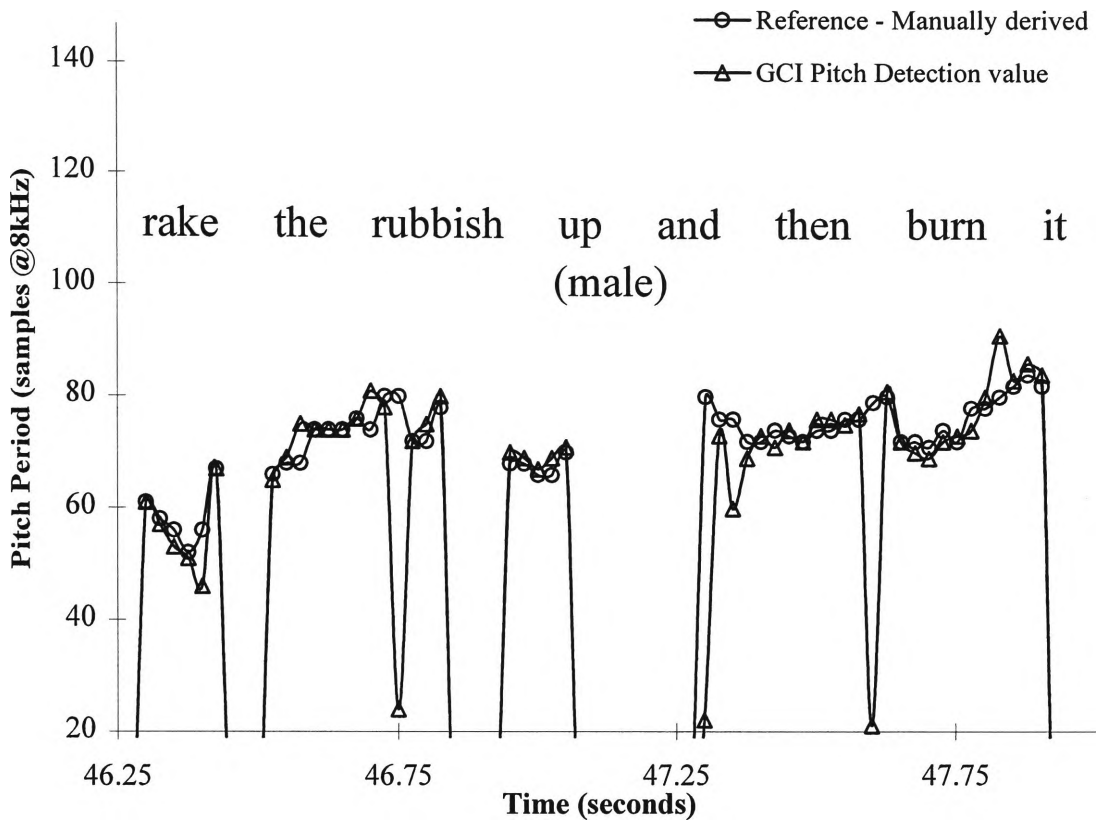

Figure B19 - GCI Pitch Detector generated Pitch Profile

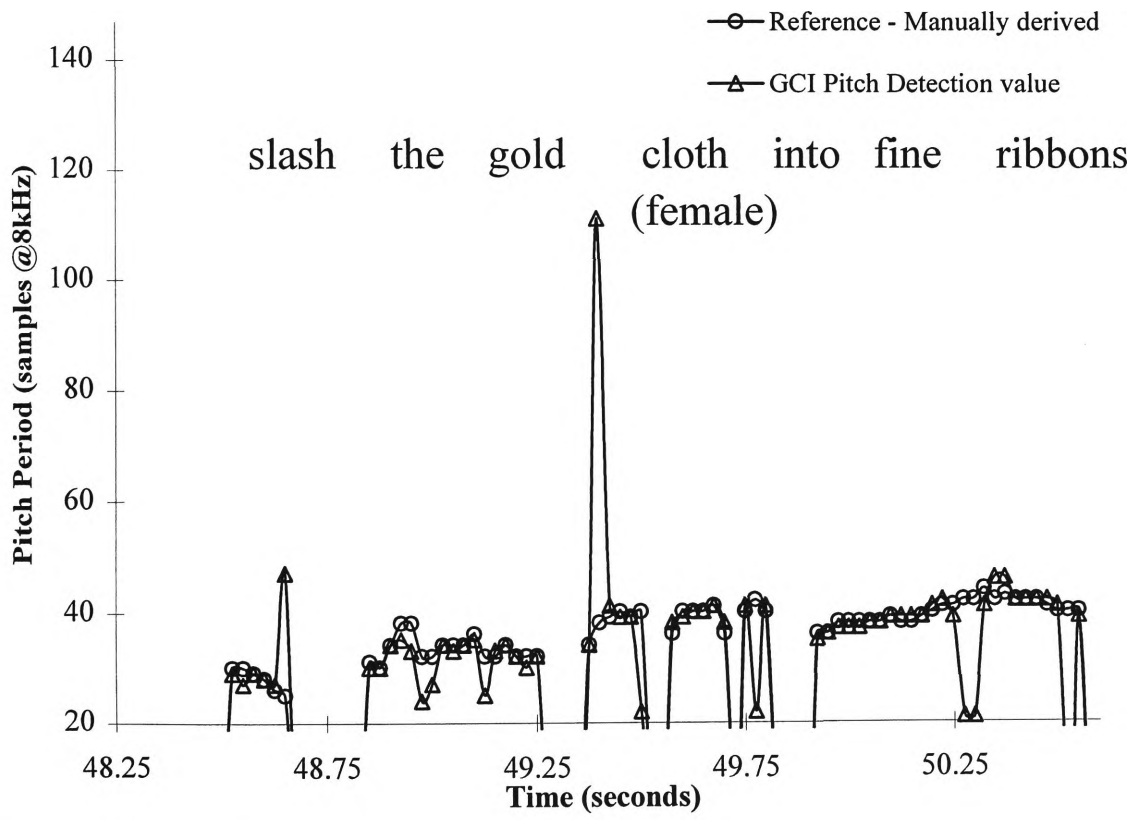

Figure B20 - GCI Pitch Detector generated Pitch Profile

APPENDIX C

Generated Pitch Profiles

using

Prototype Waveform Pitch Detection

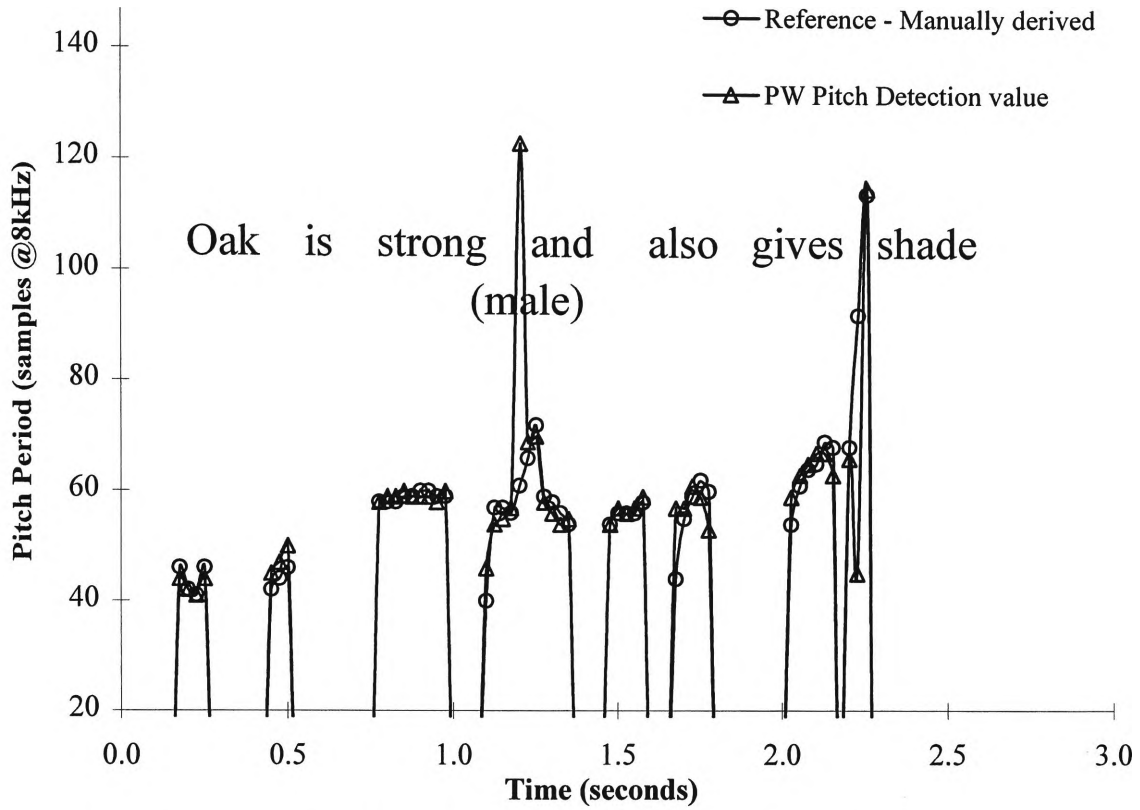

Figure C1 - Prototype Waveform Pitch Detector generated Pitch Profile

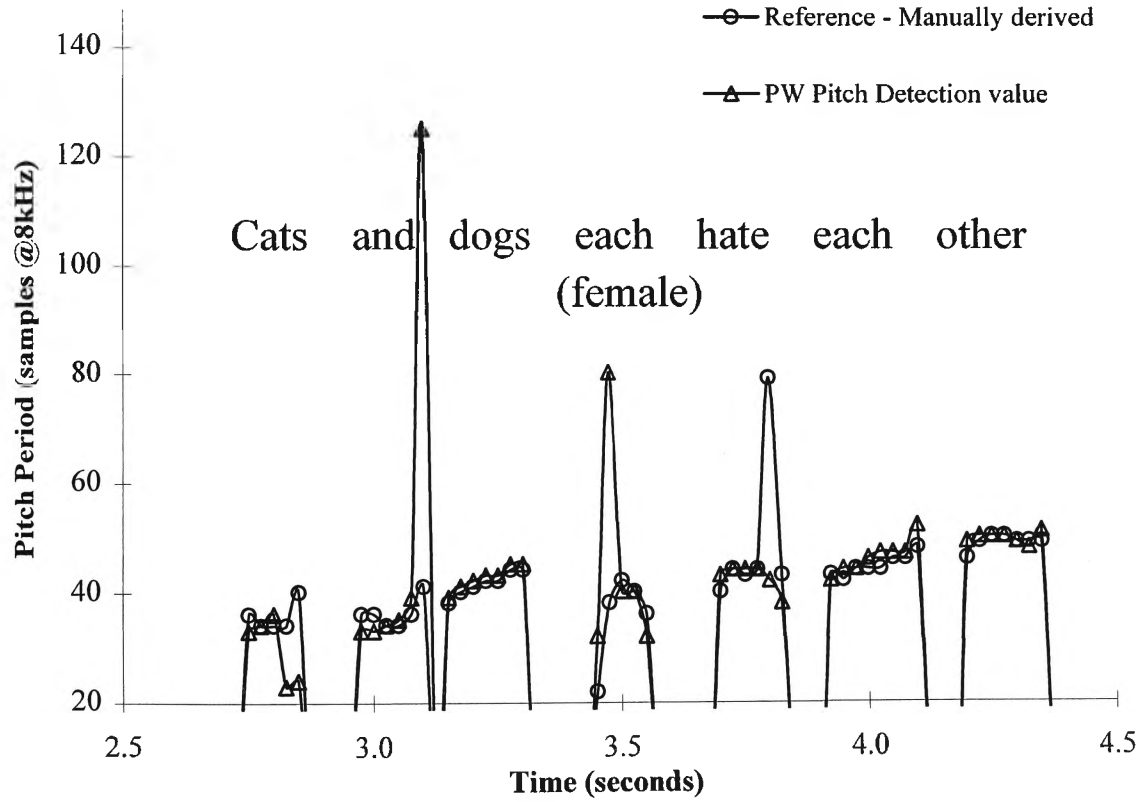

Figure C2 - Prototype Waveform Pitch Detector generated Pitch Profile

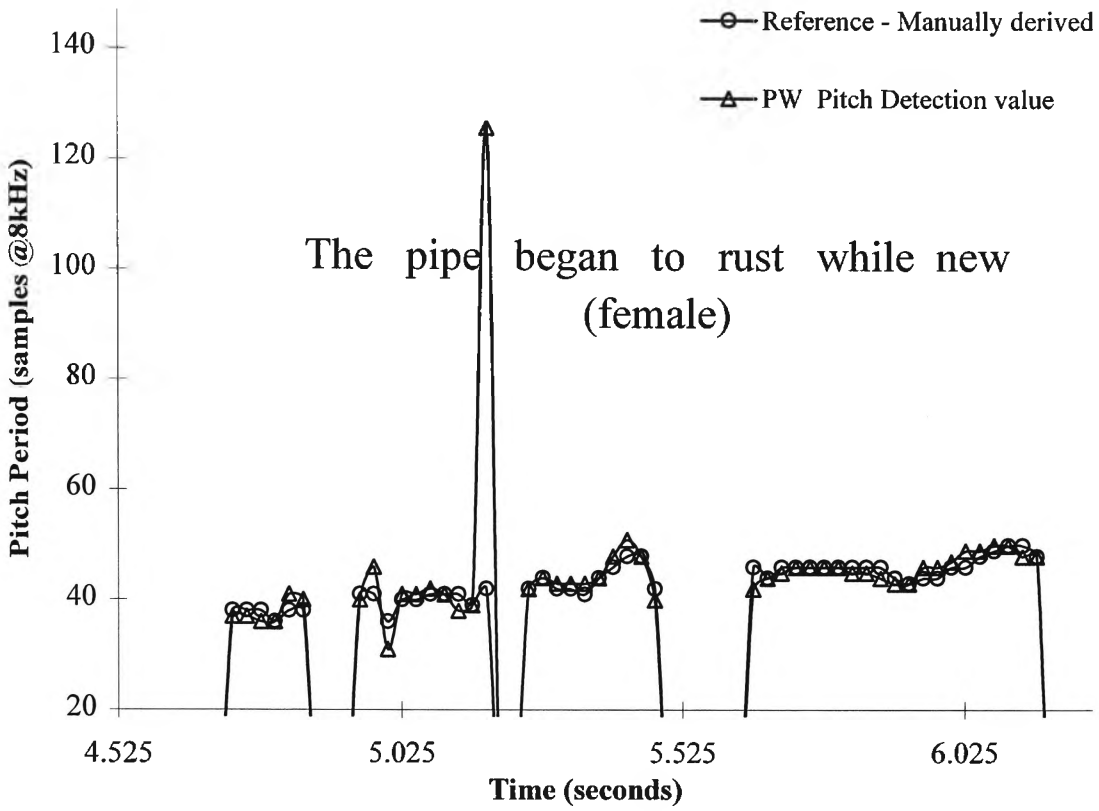

Figure C3 - Prototype Waveform Pitch Detector generated Pitch Profile

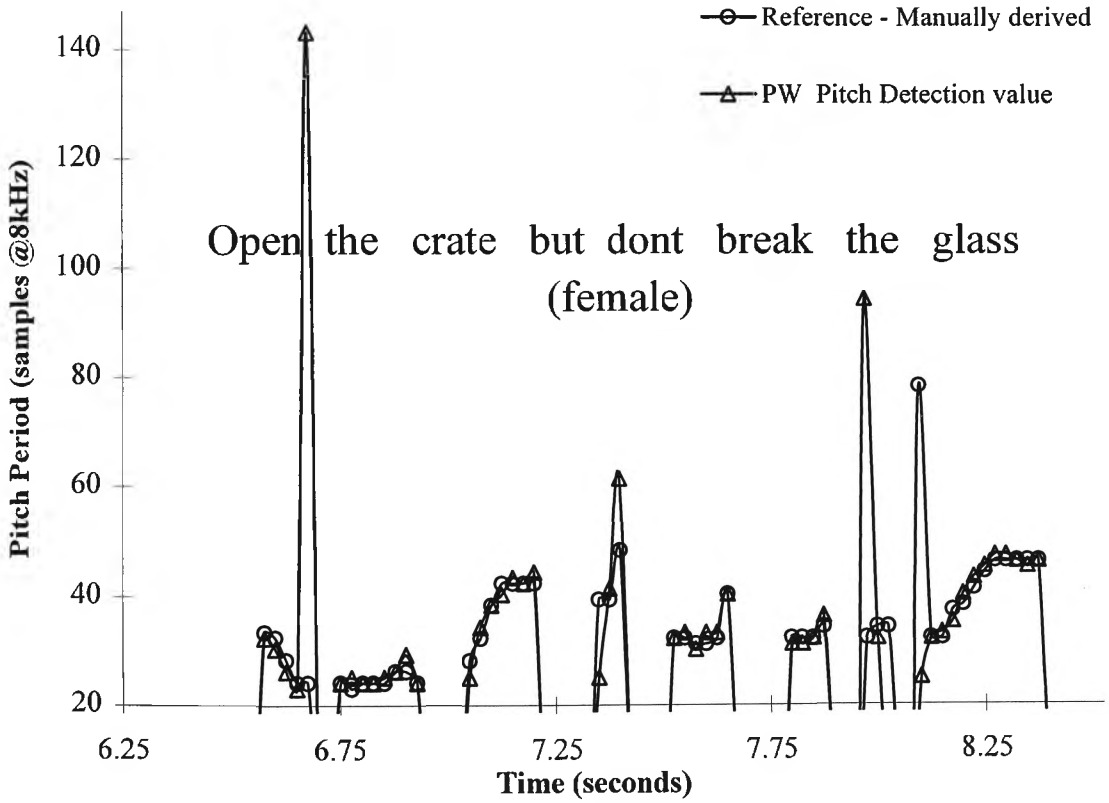

Figure C4 - Prototype Waveform Pitch Detector generated Pitch Profile

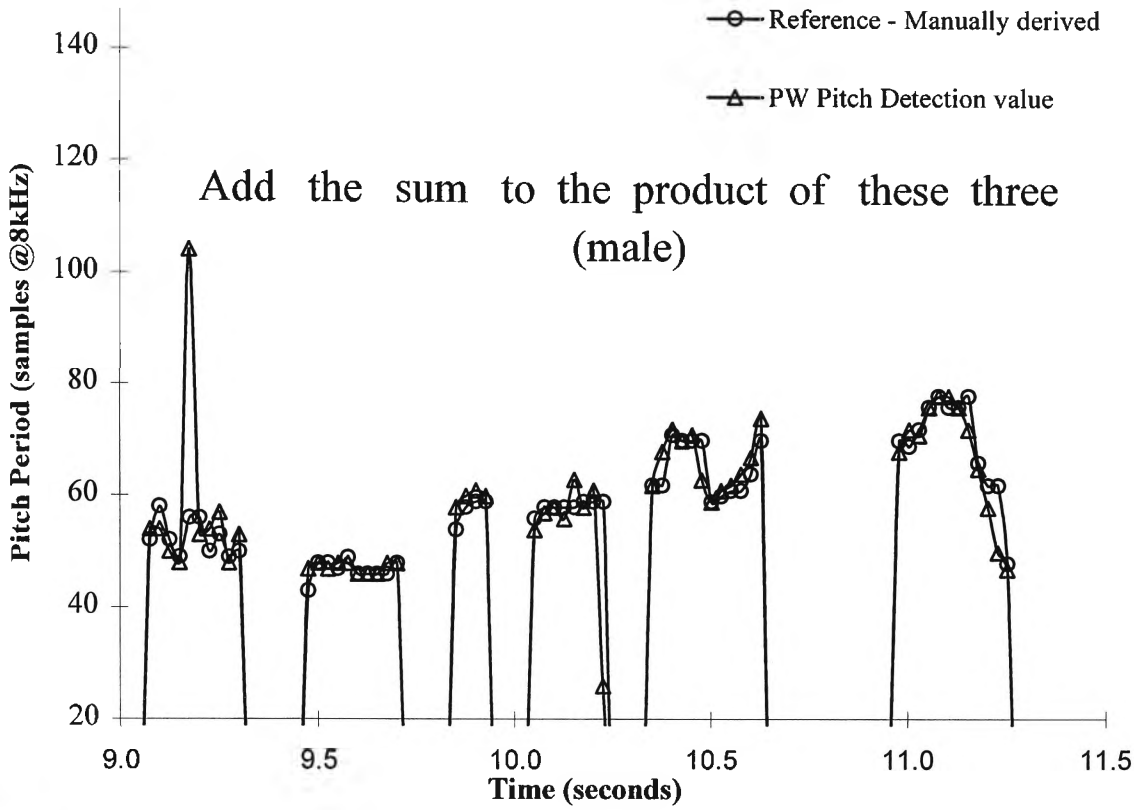

Figure C5 - Prototype Waveform Pitch Detector generated Pitch Profile

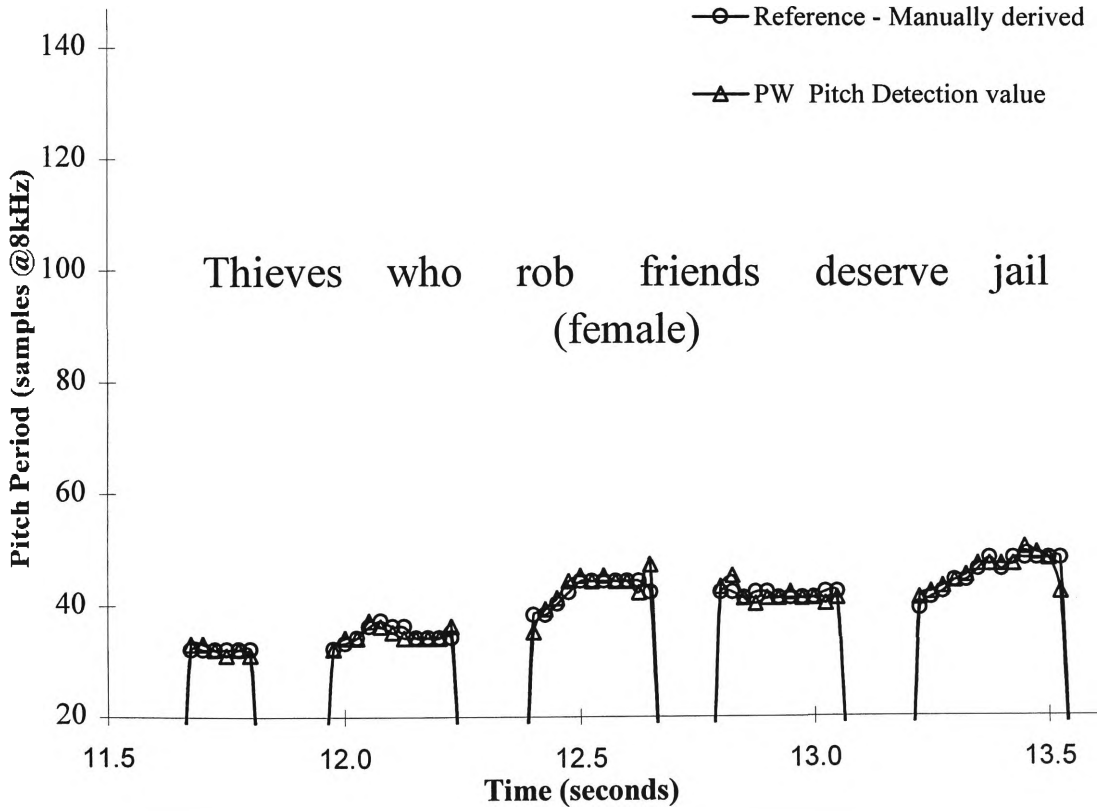

Figure C6 - Prototype Waveform Pitch Detector generated Pitch Profile

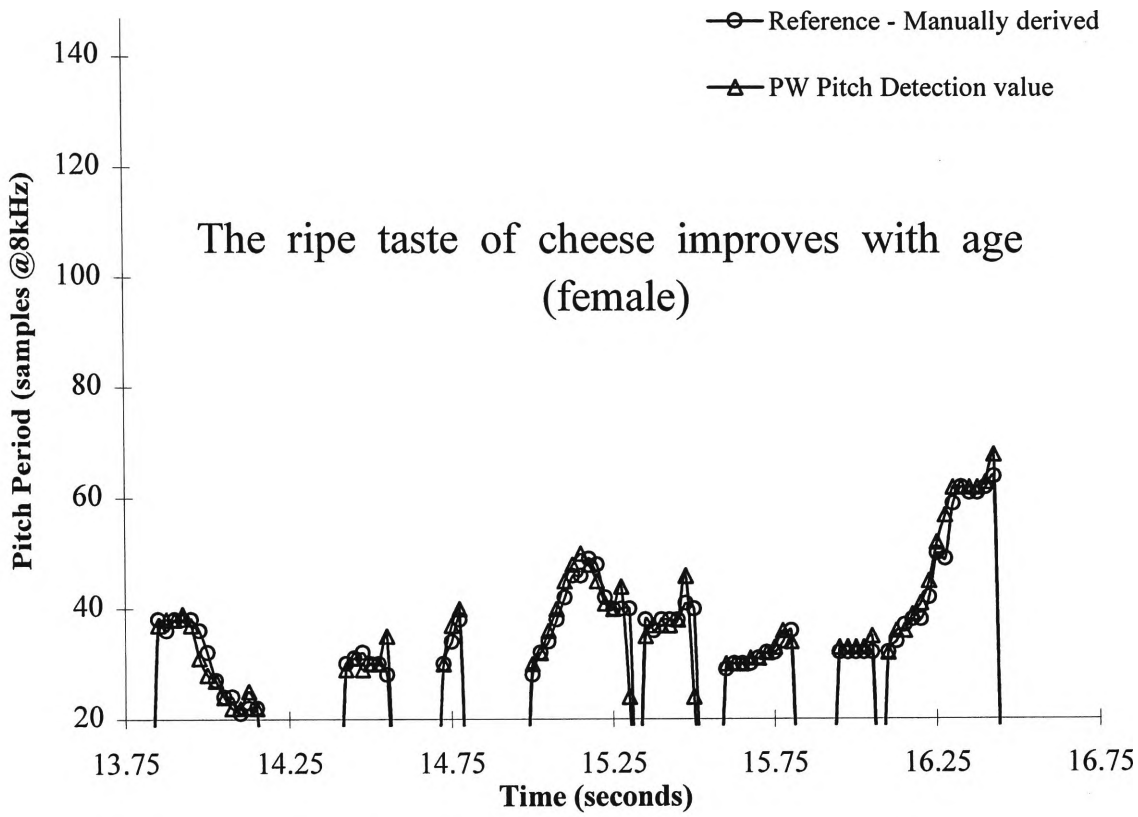

Figure C7 - Prototype Waveform Pitch Detector generated Pitch Profile

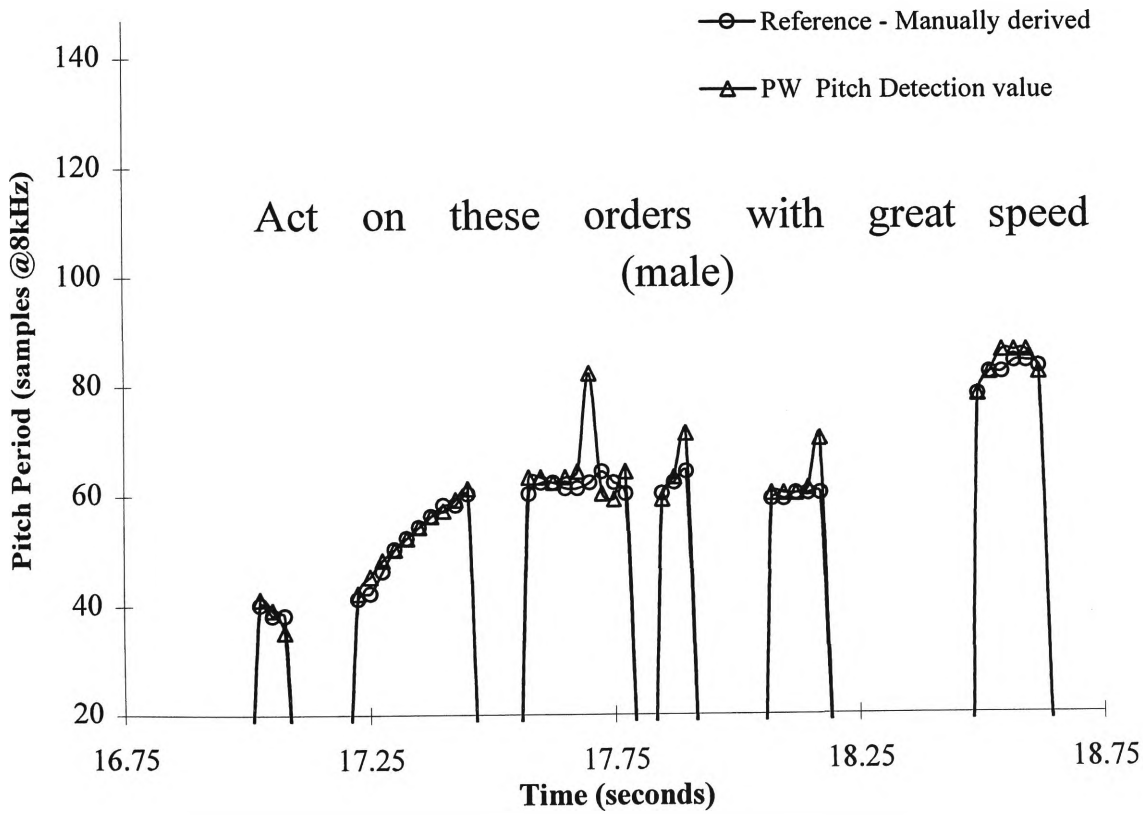

Figure C8 - Prototype Waveform Pitch Detector generated Pitch Profile

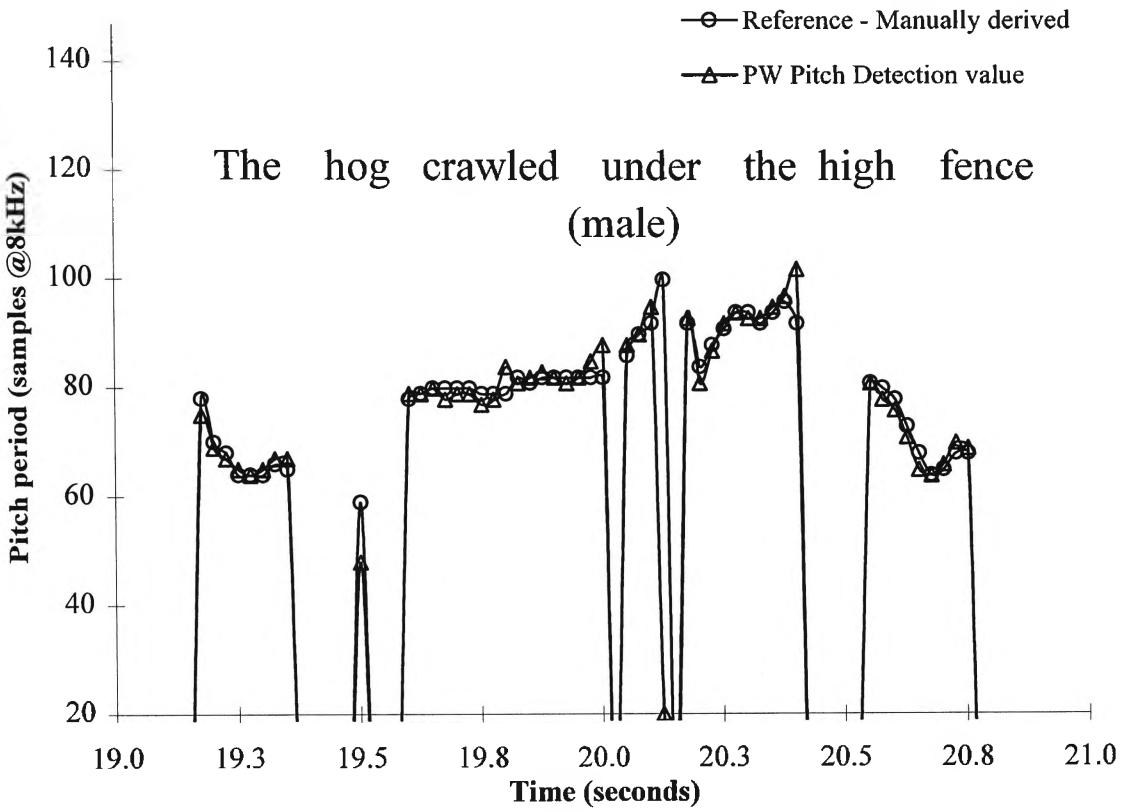

Figure C9 - Prototype Waveform Pitch Detector generated Pitch Profile

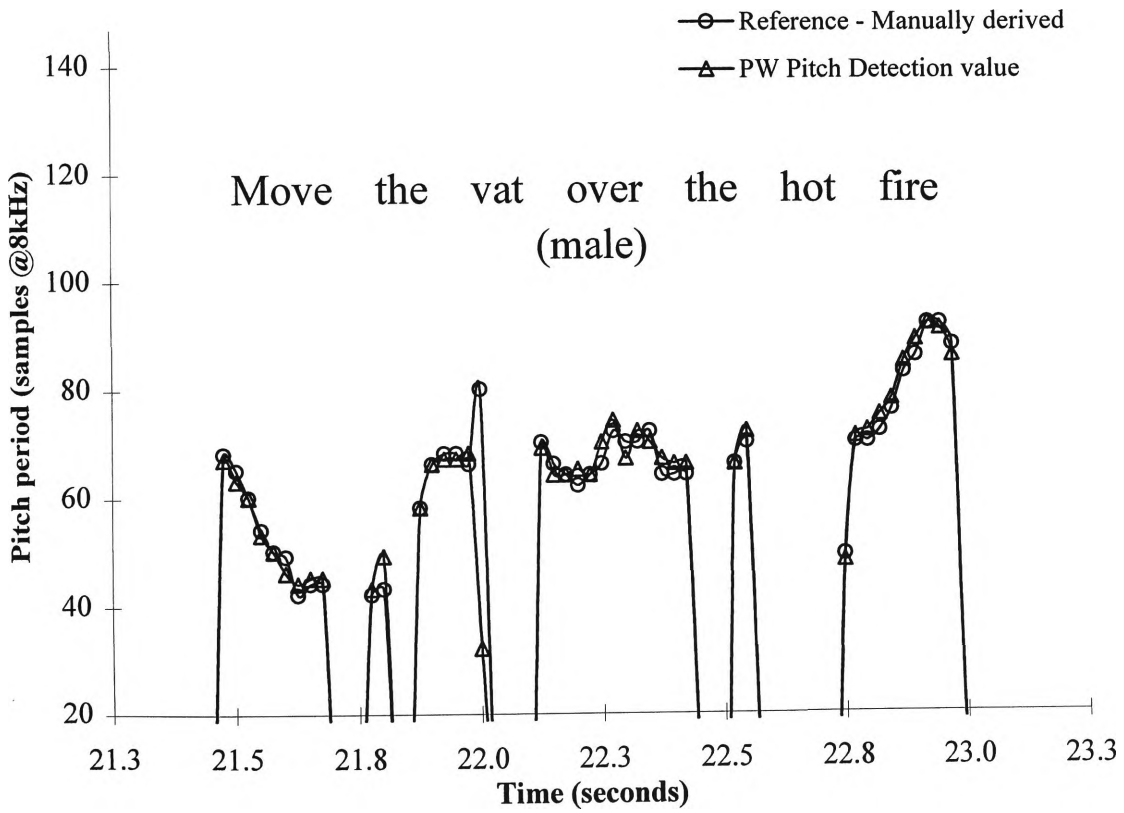

Figure C10 - Prototype Waveform Pitch Detector generated Pitch Profile

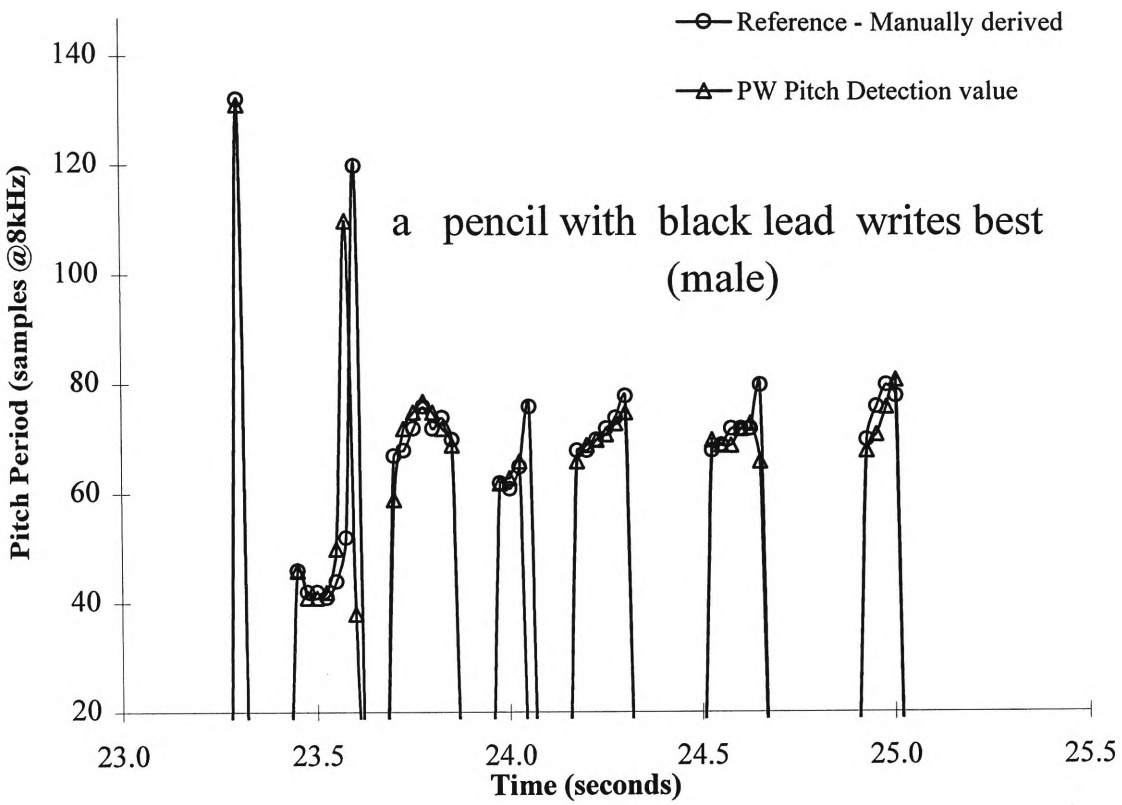

Figure C11 - Prototype Waveform Pitch Detector generated Pitch Profile

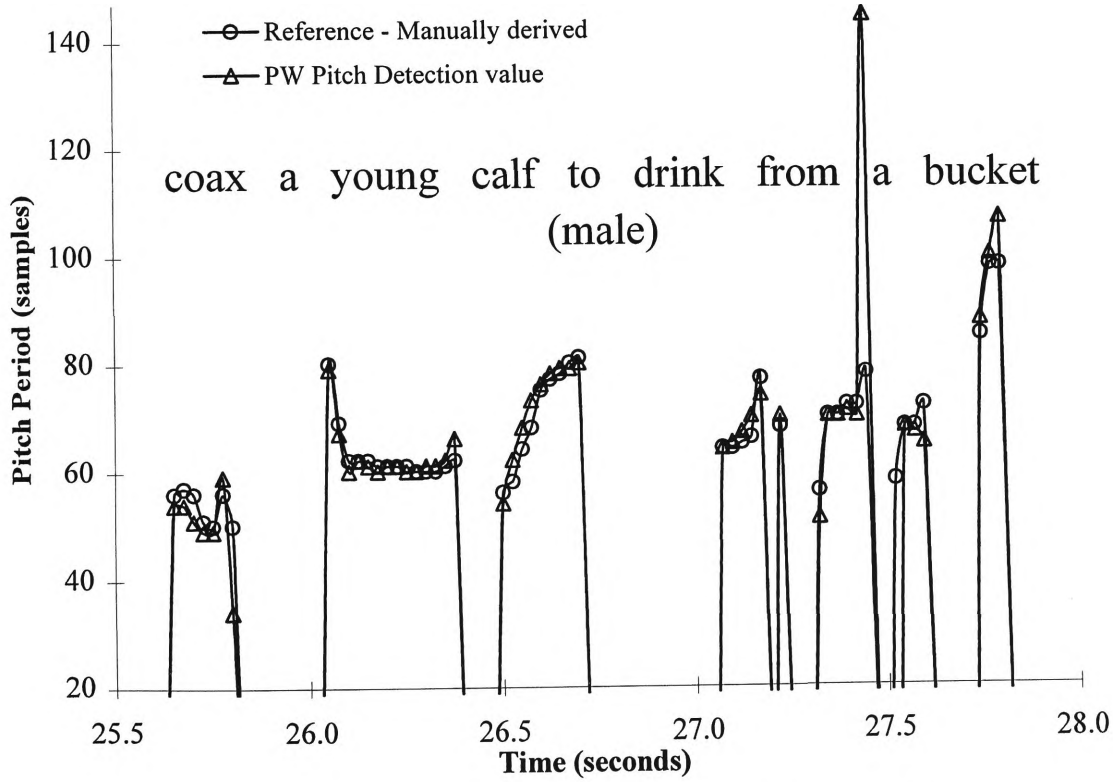

Figure C12 - Prototype Waveform Pitch Detector generated Pitch Profile

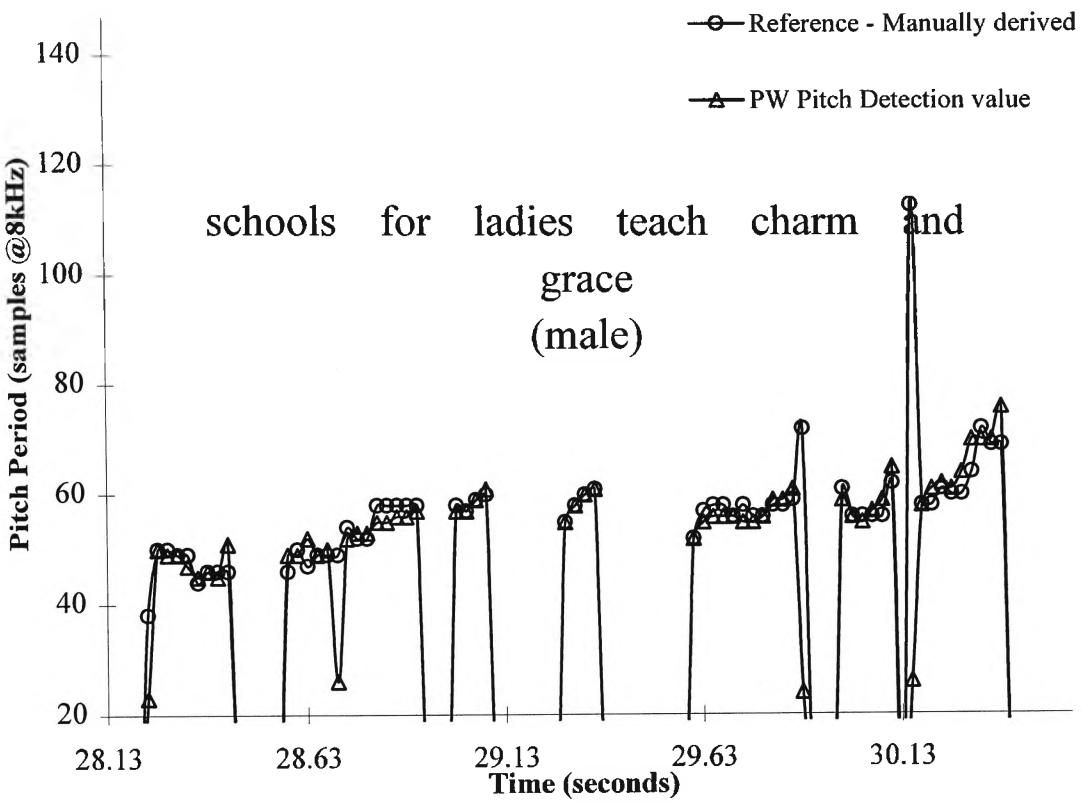

Figure C13 - Prototype Waveform Pitch Detector generated Pitch Profile

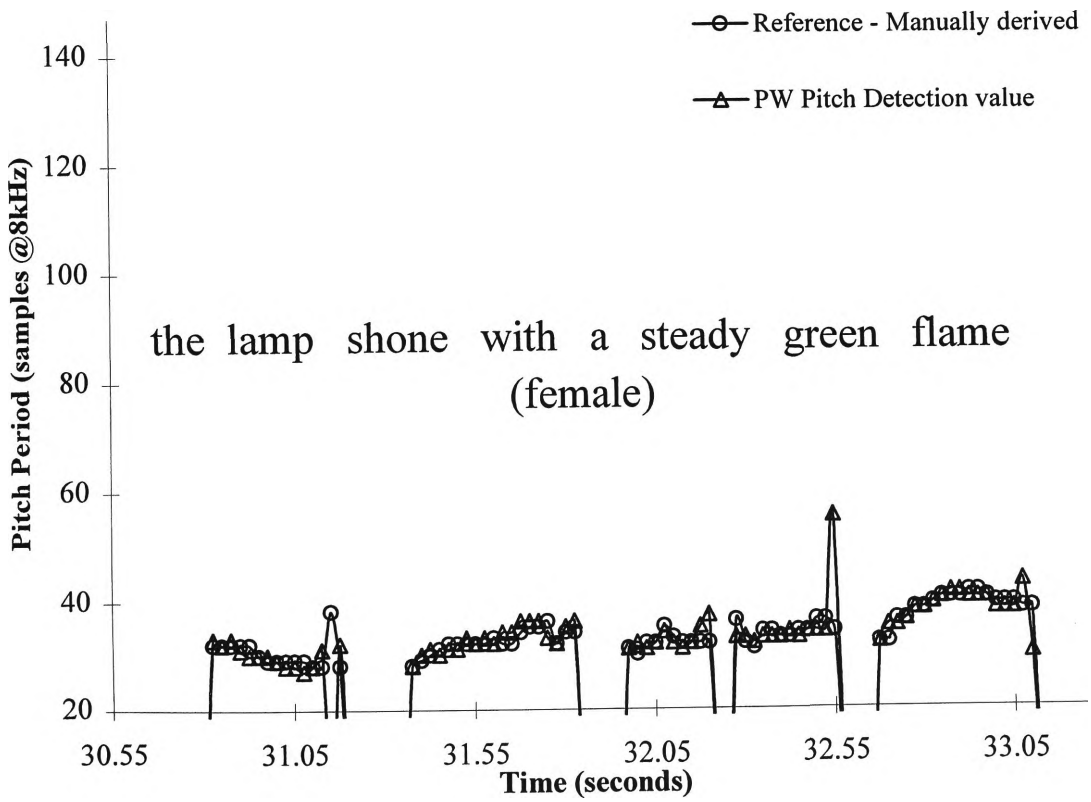

Figure C14 - Prototype Waveform Pitch Detector generated Pitch Profile

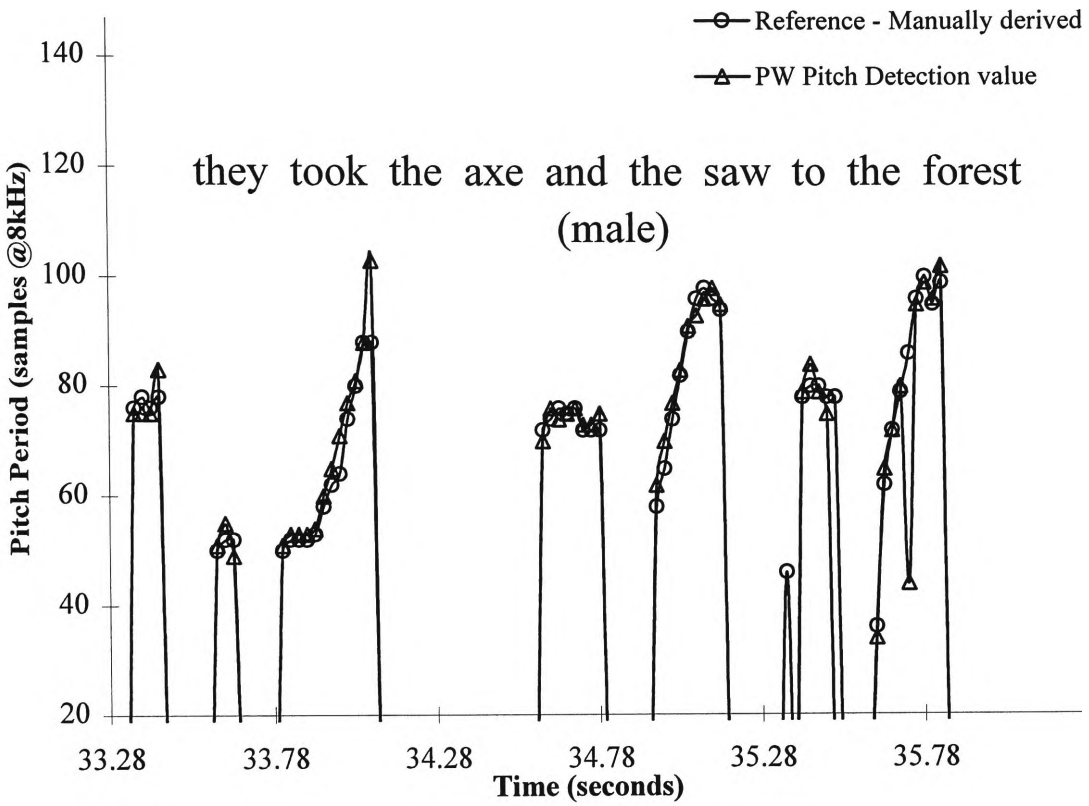

Figure C15 - Prototype Waveform Pitch Detector generated Pitch Profile

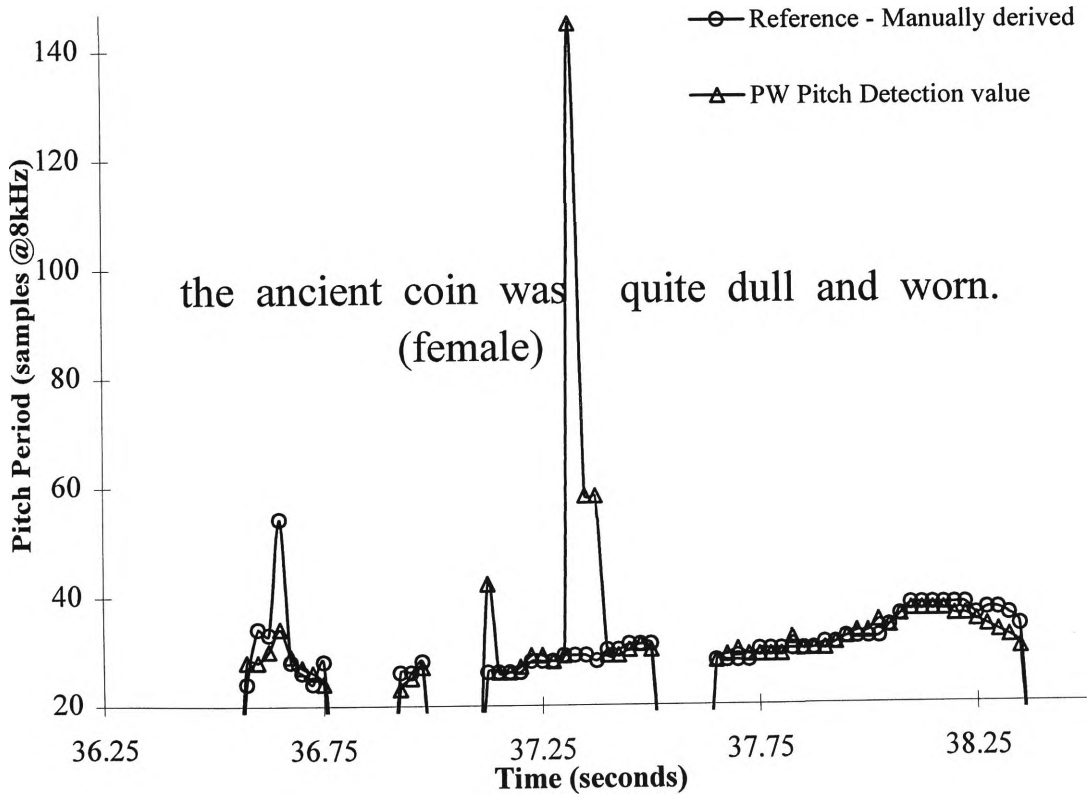

Figure C16 - Prototype Waveform Pitch Detector generated Pitch Profile

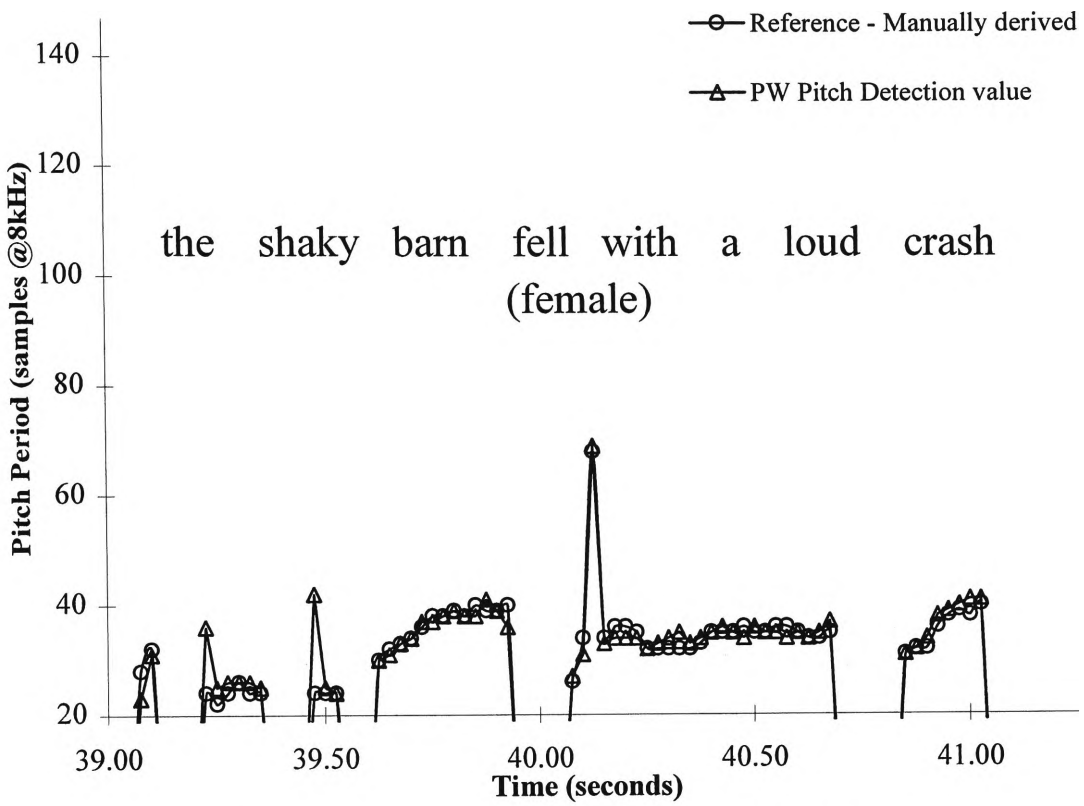

Figure C17 - Prototype Waveform Pitch Detector generated Pitch Profile

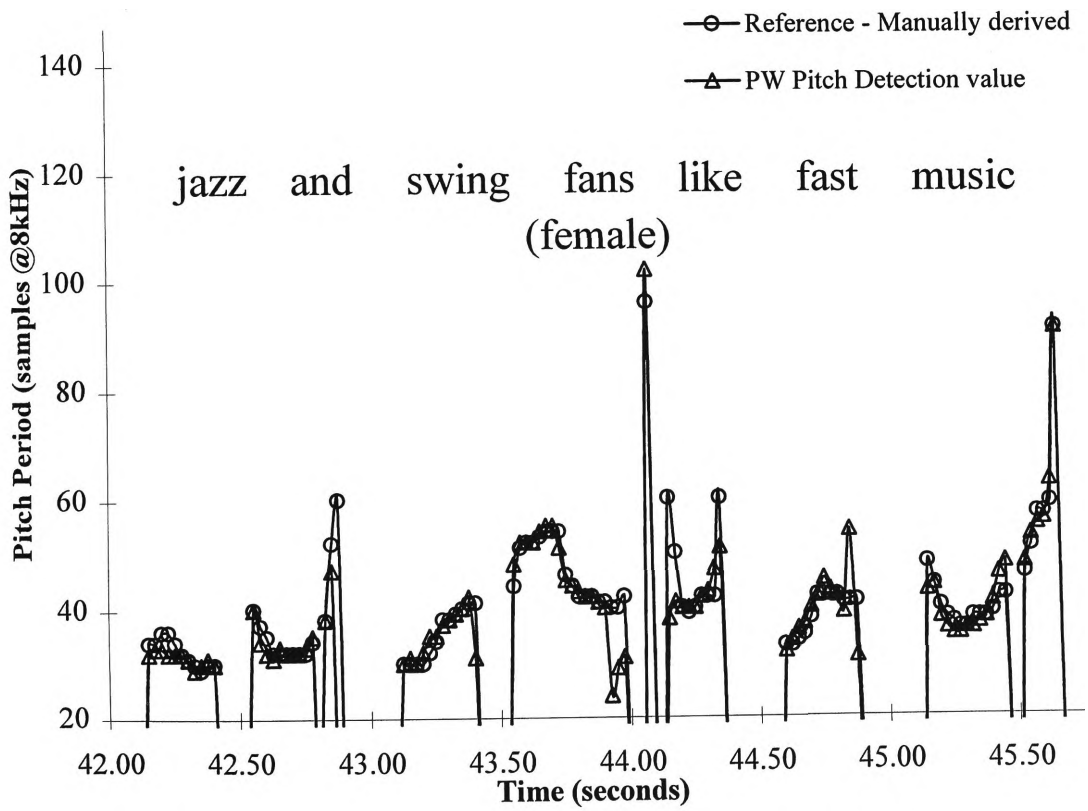

Figure C18 - Prototype Waveform Pitch Detector generated Pitch Profile

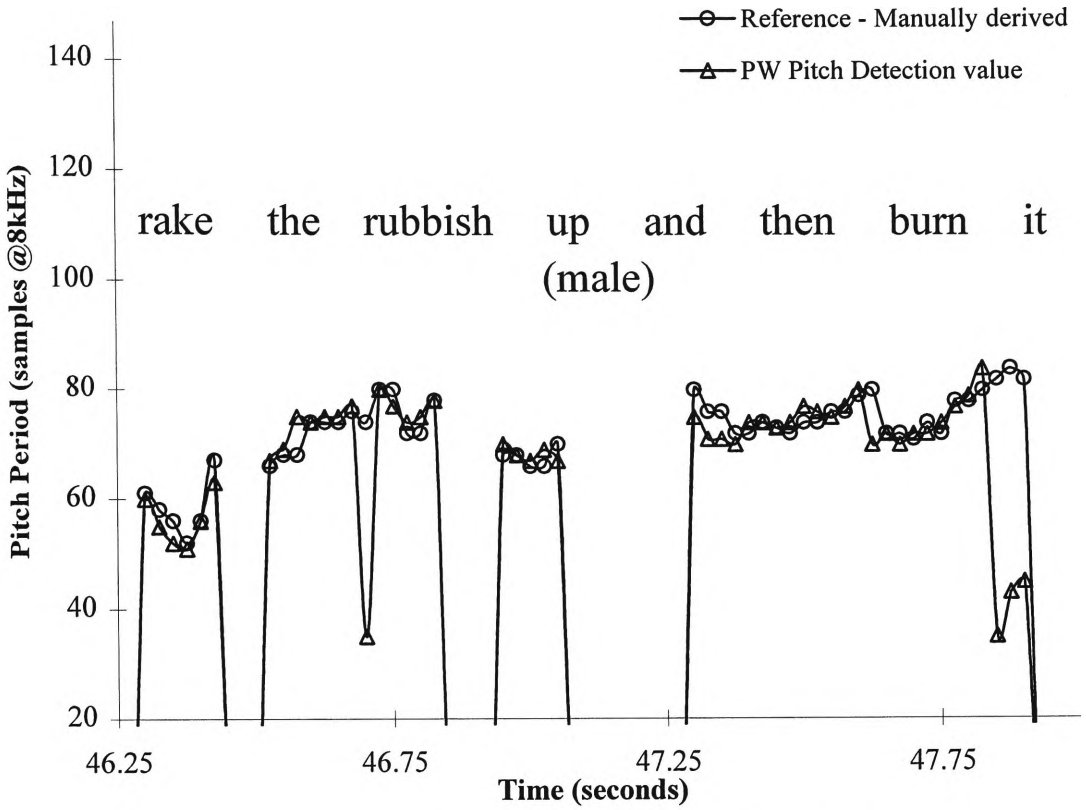

Figure C19 - Prototype Waveform Pitch Detector generated Pitch Profile

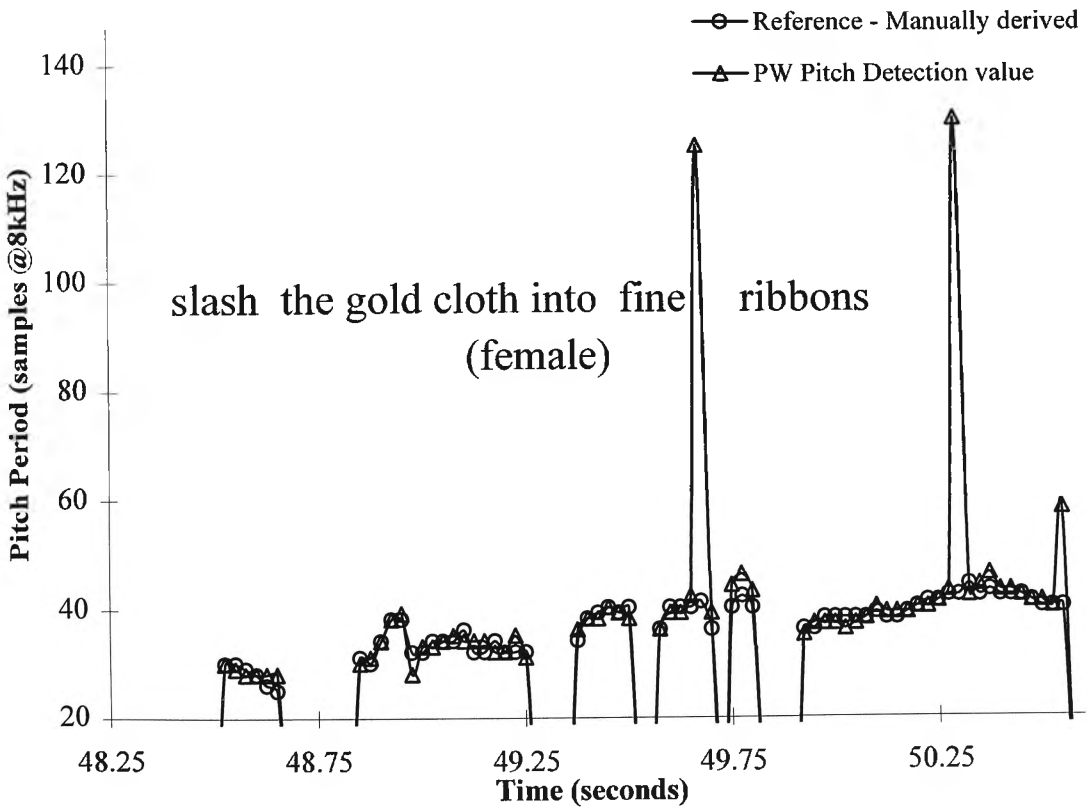

Figure C20 - Prototype Waveform Pitch Detector generated Pitch Profile

APPENDIX D

Generated Pitch Profiles

using

Dynamic Programming/Viterbi Pitch Detection

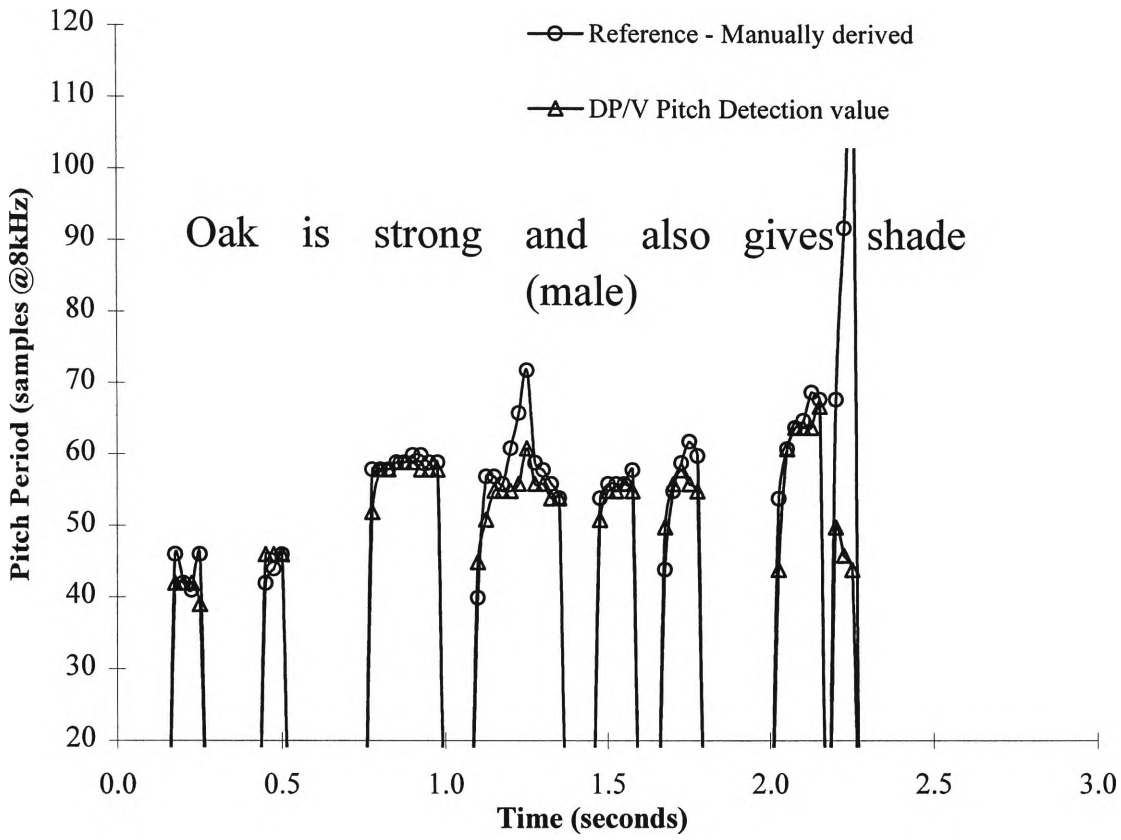

Figure D1 - DP/V Pitch Detector generated Pitch Profile

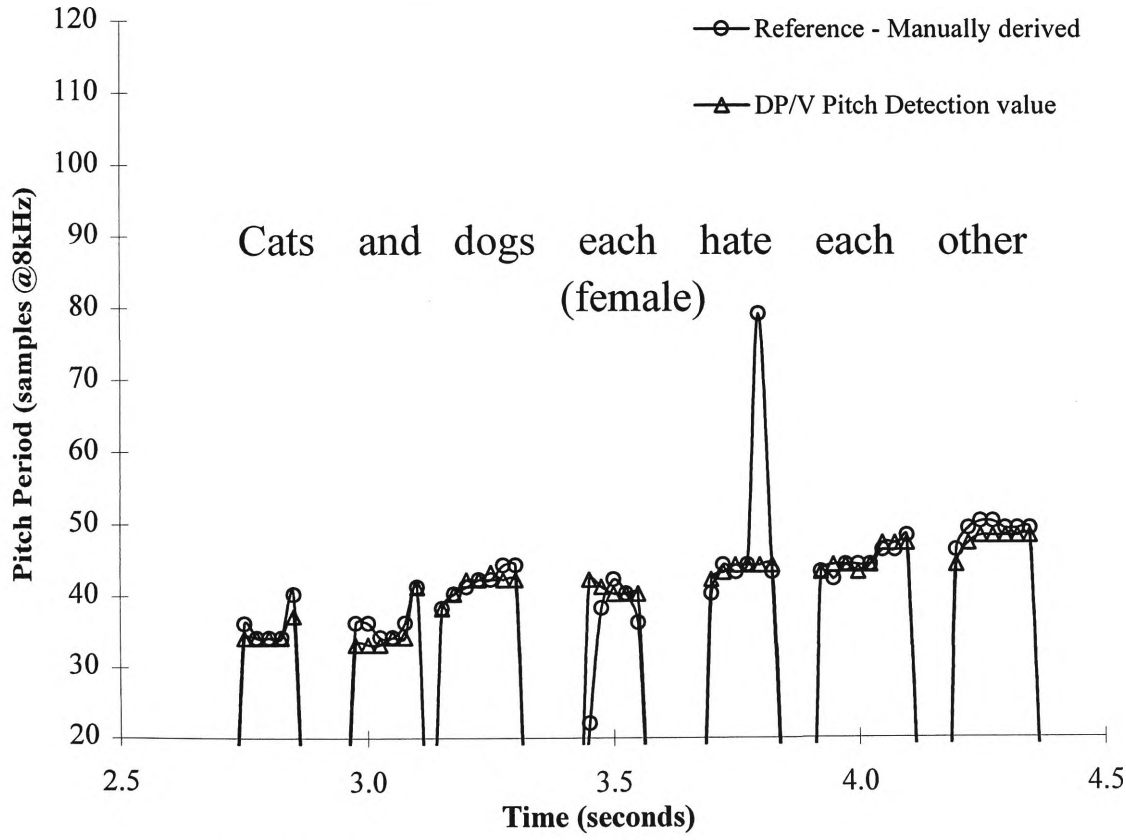

Figure D2 - DP/V Pitch Detector generated Pitch Profile

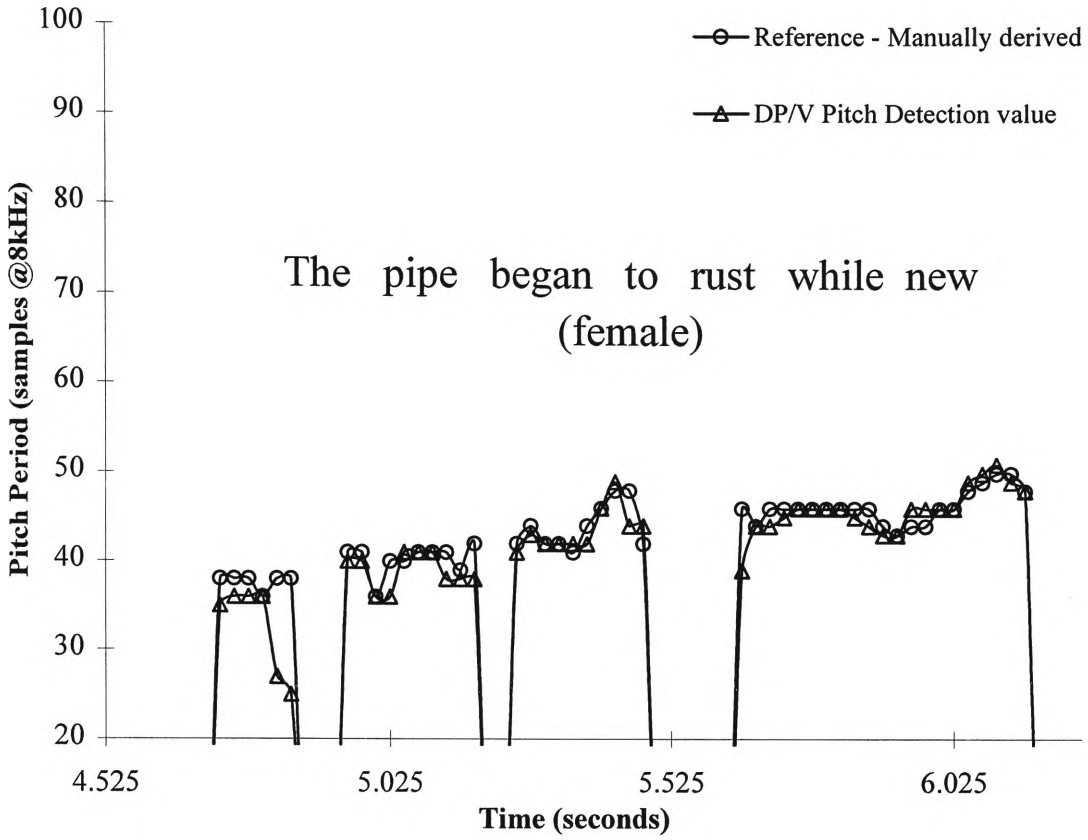

Figure D3 - DP/V Pitch Detector generated Pitch Profile

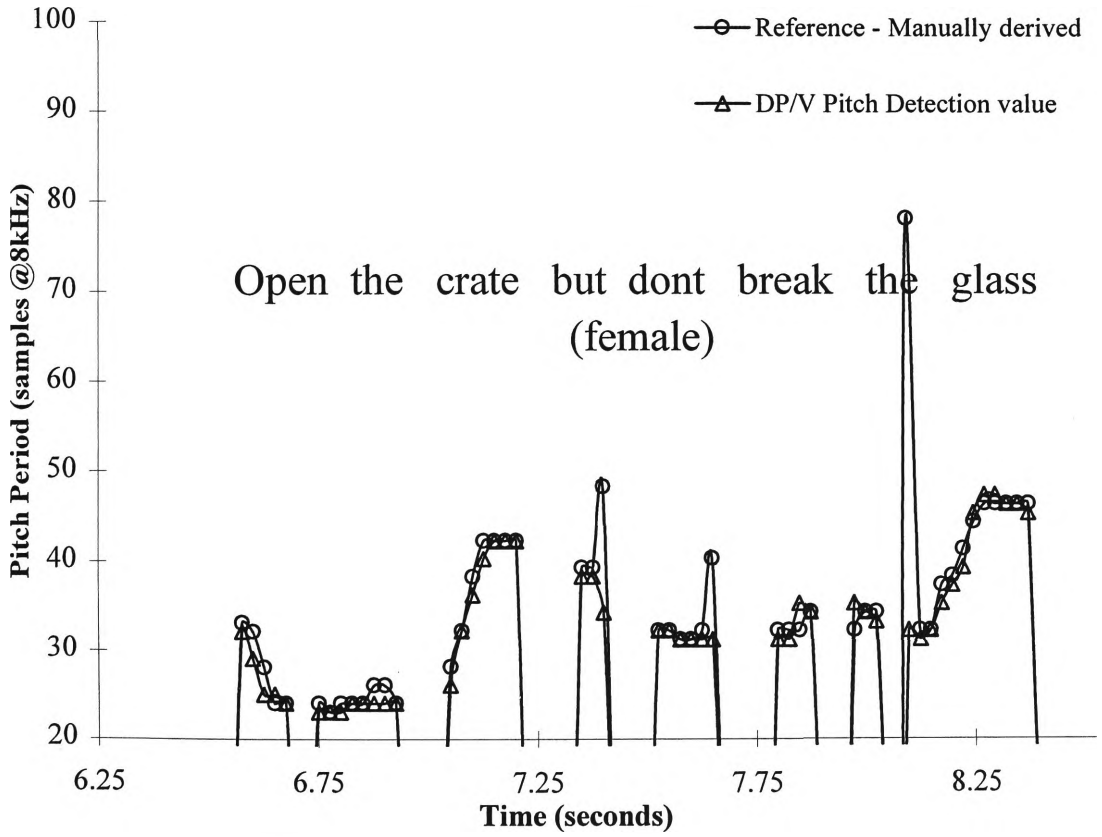

Figure D4 - DP/V Pitch Detector generated Pitch Profile

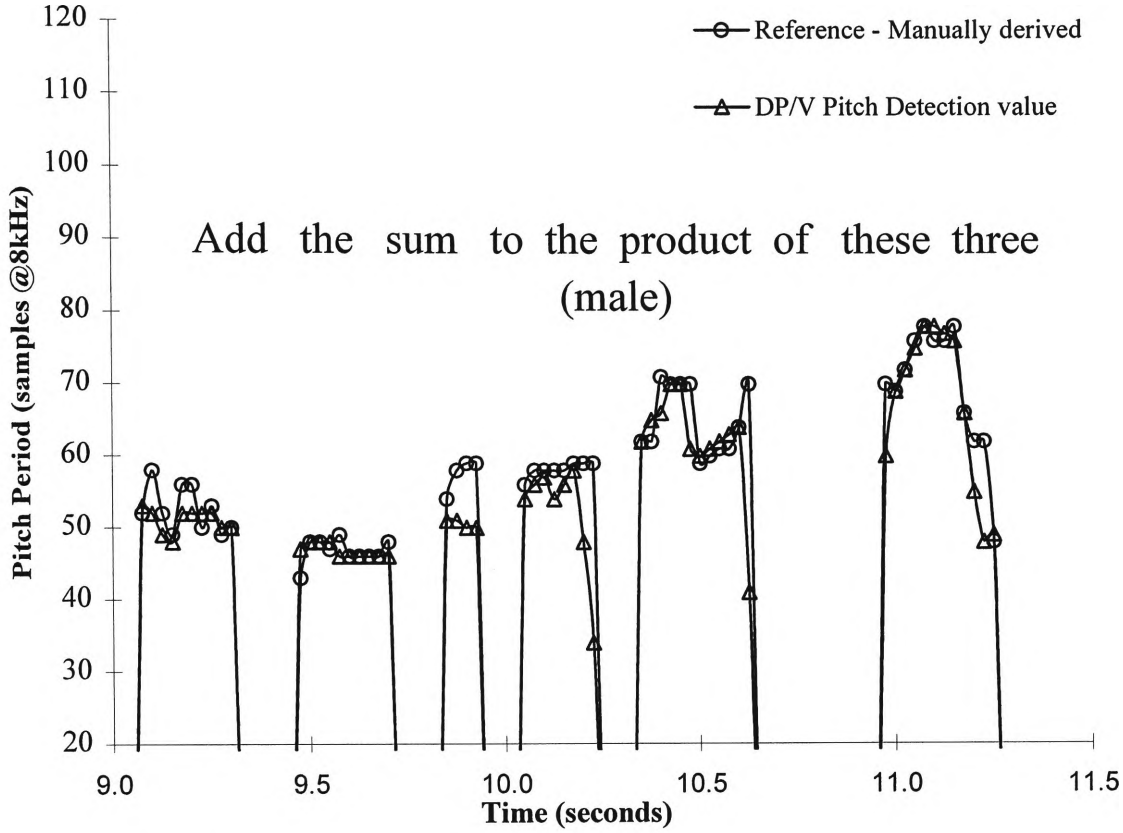

Figure D5 - DP/V Pitch Detector generated Pitch Profile

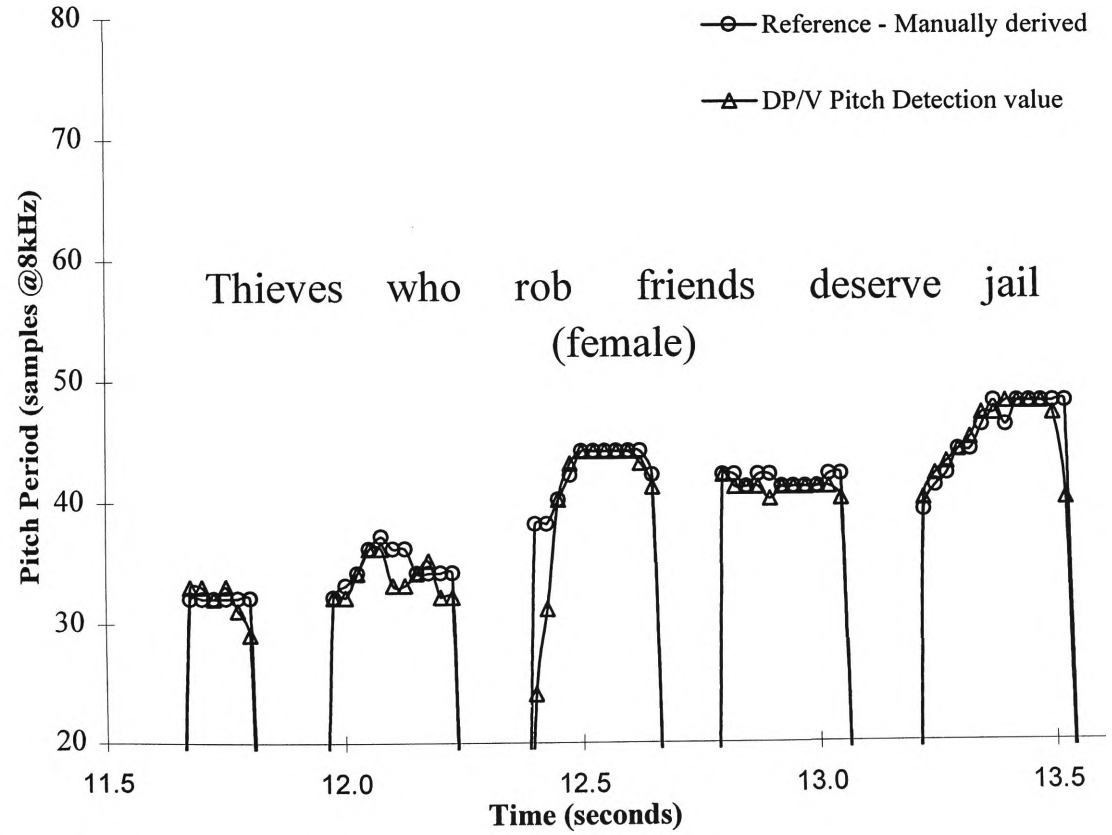

Figure D6 - DP/V Pitch Detector generated Pitch Profile

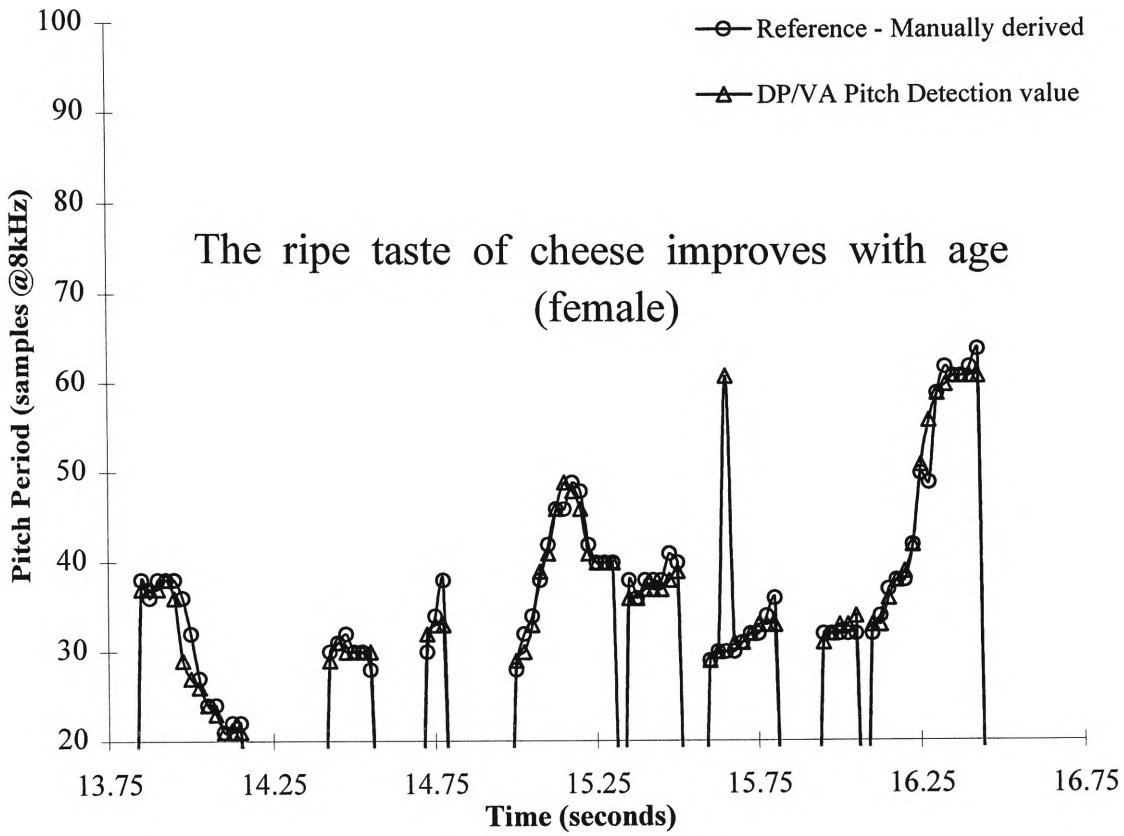

Figure D7 - DP/V Pitch Detector generated Pitch Profile

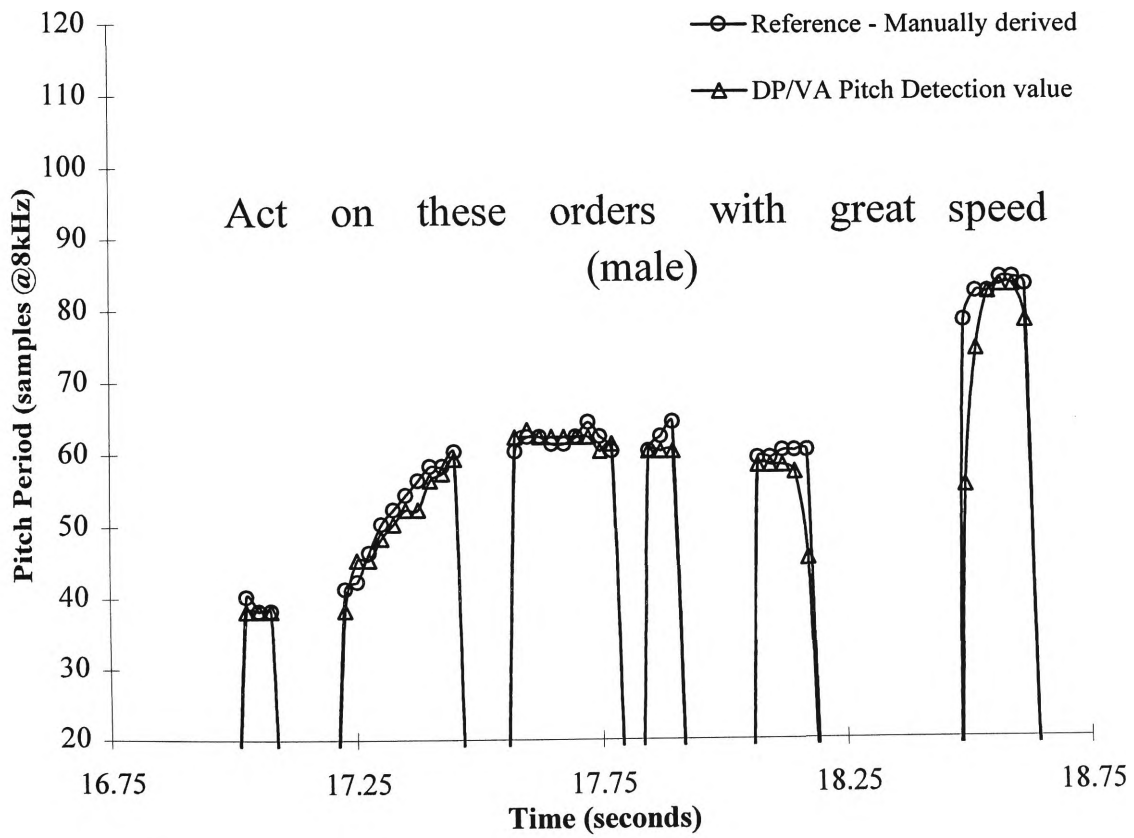

Figure D8 - DP/V Pitch Detector generated Pitch Profile

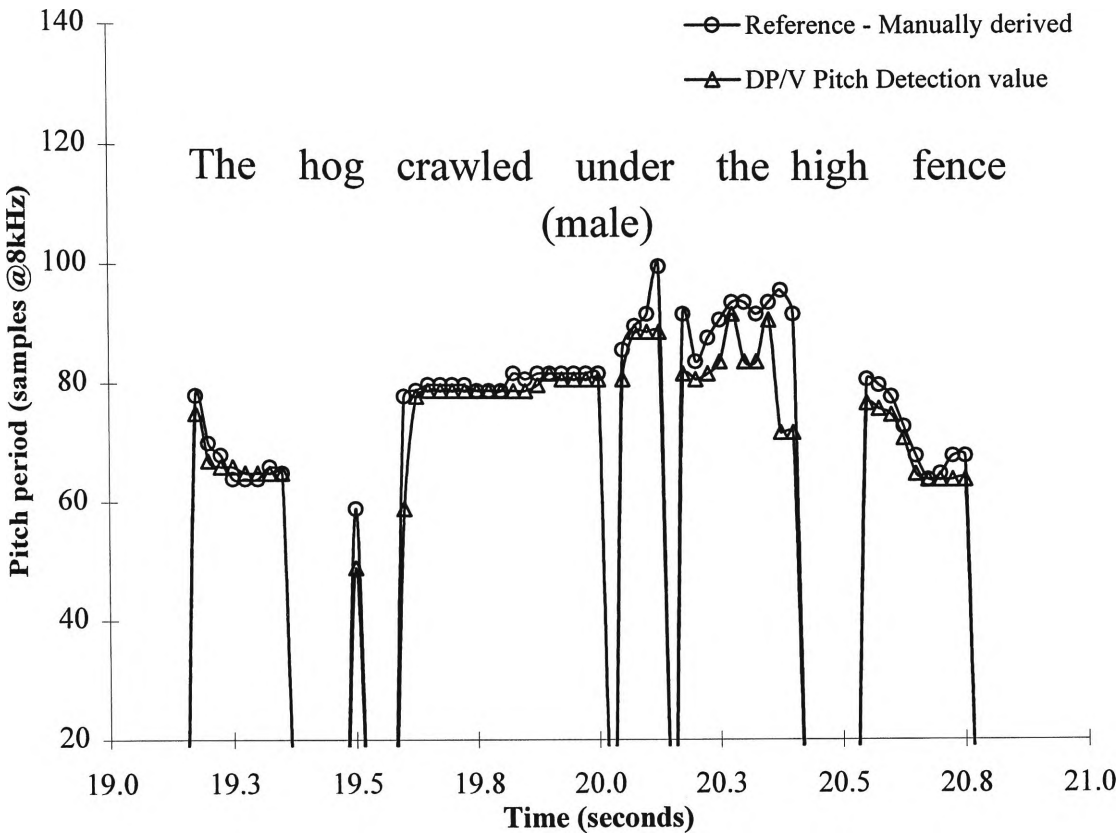

Figure D9 - DP/V Pitch Detector generated Pitch Profile

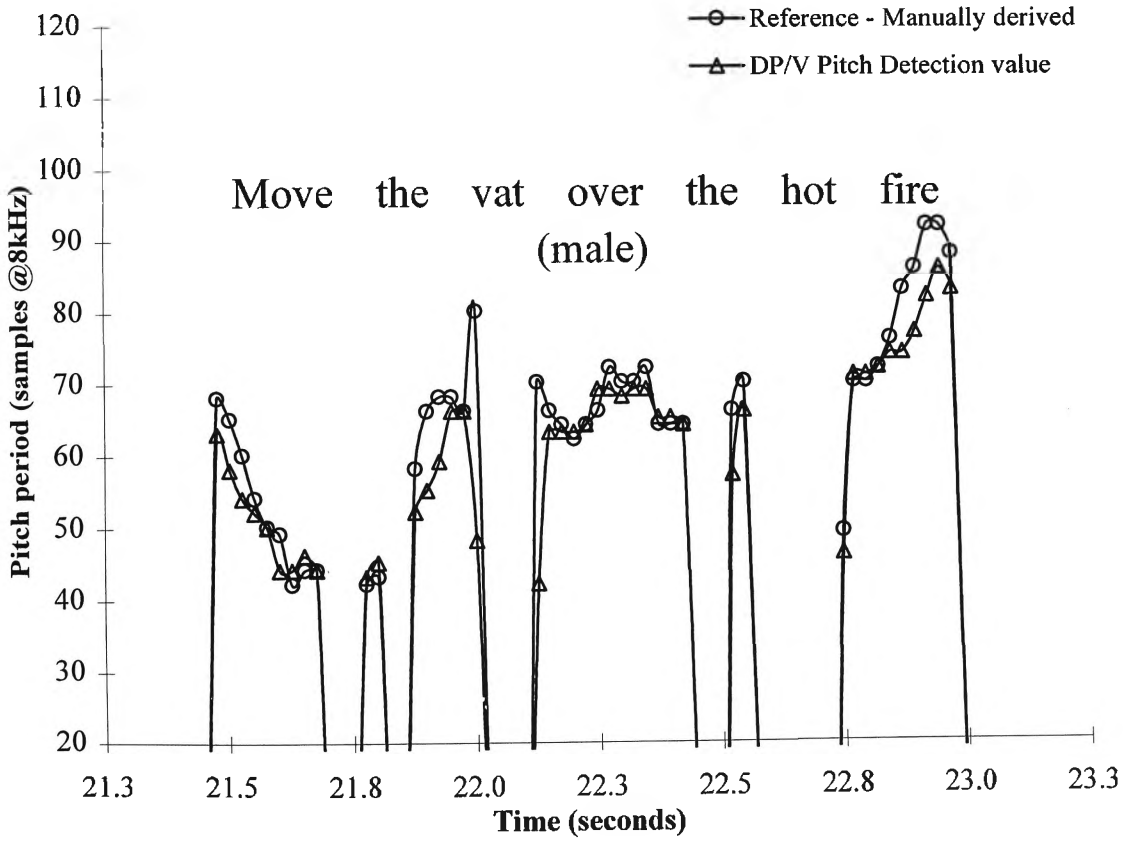

Figure D10 - DP/V Pitch Detector generated Pitch Profile

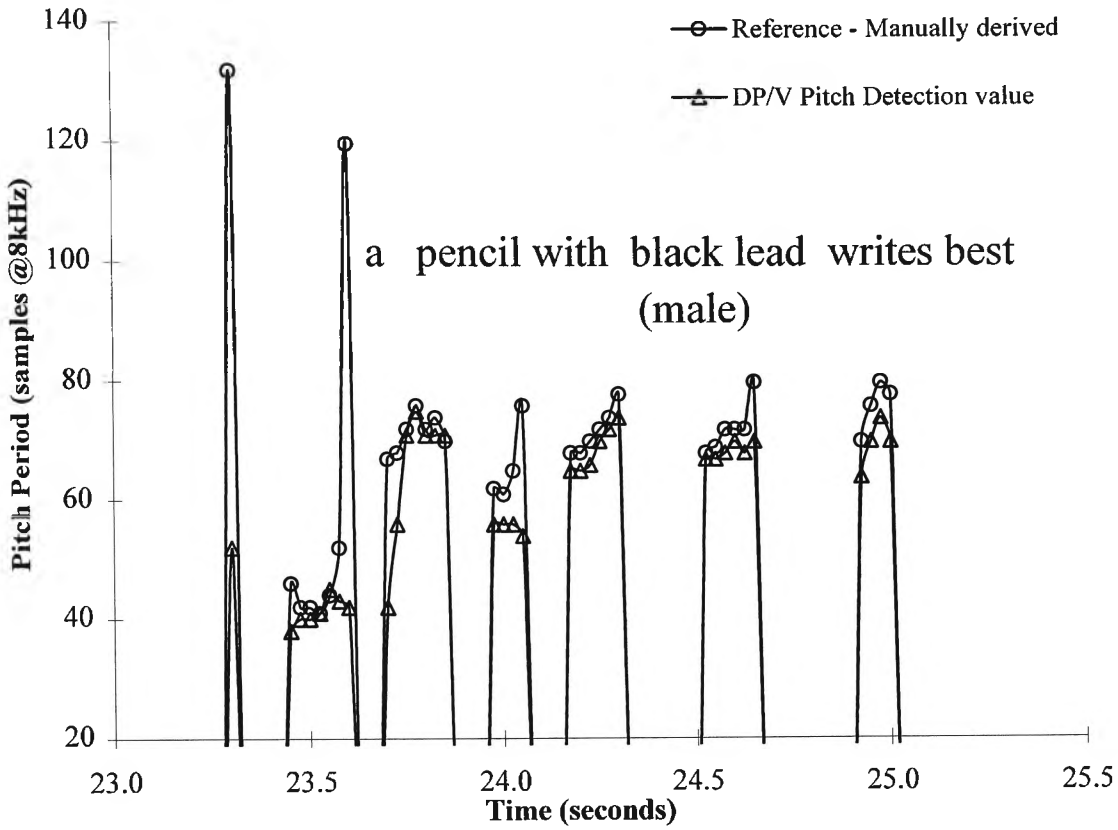

Figure D11 - DP/V Pitch Detector generated Pitch Profile

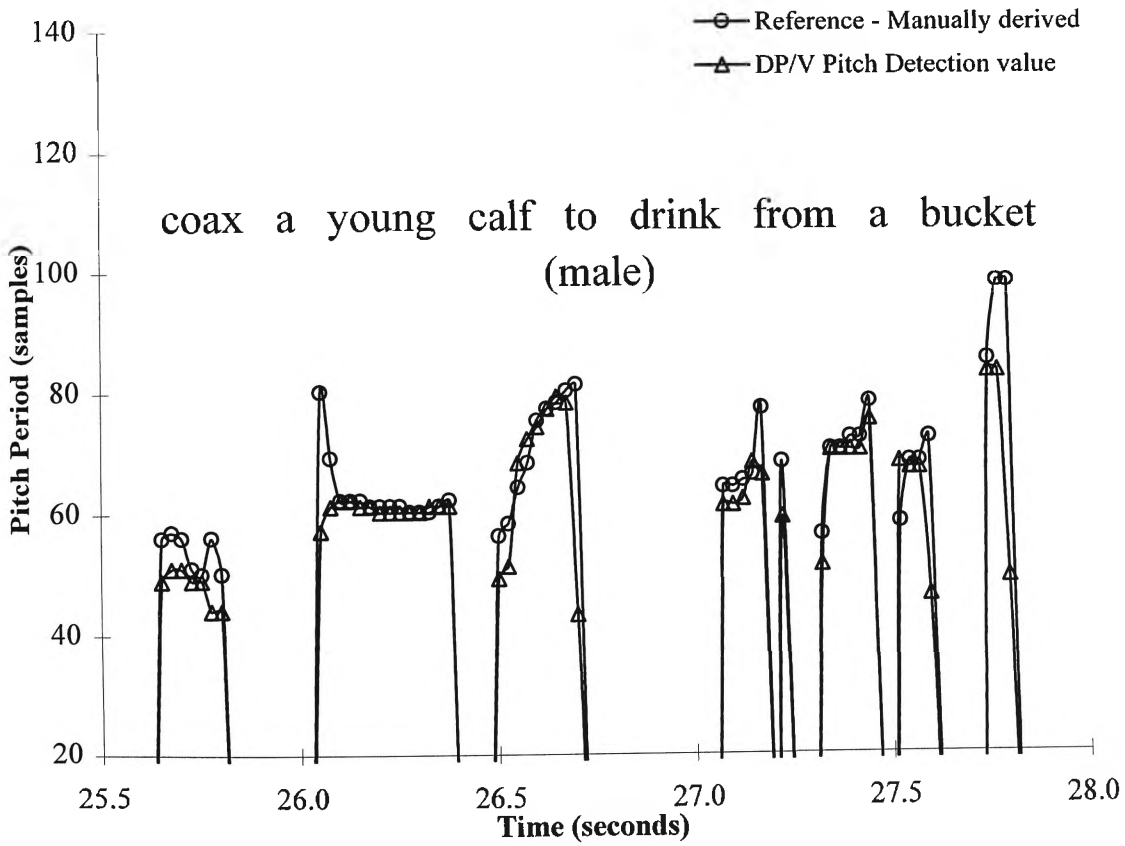

Figure D12 - DP/V Pitch Detector generated Pitch Profile

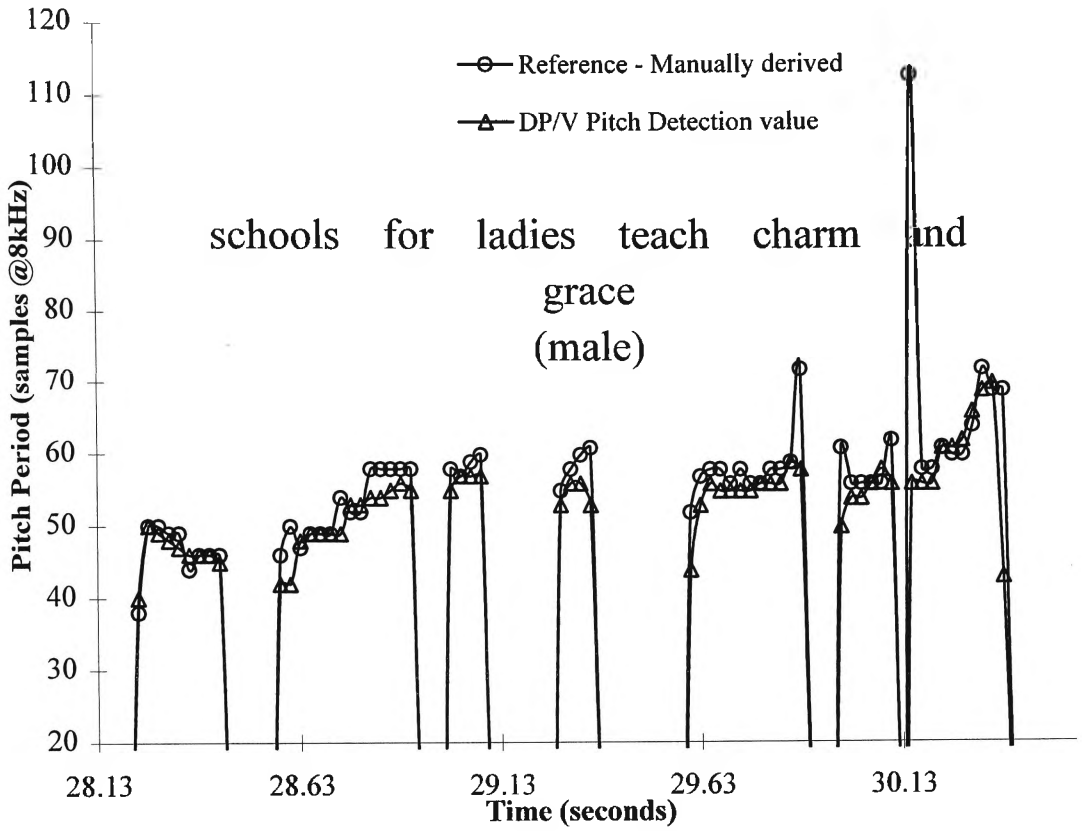

Figure D13 - DP/V Pitch Detector generated Pitch Profile

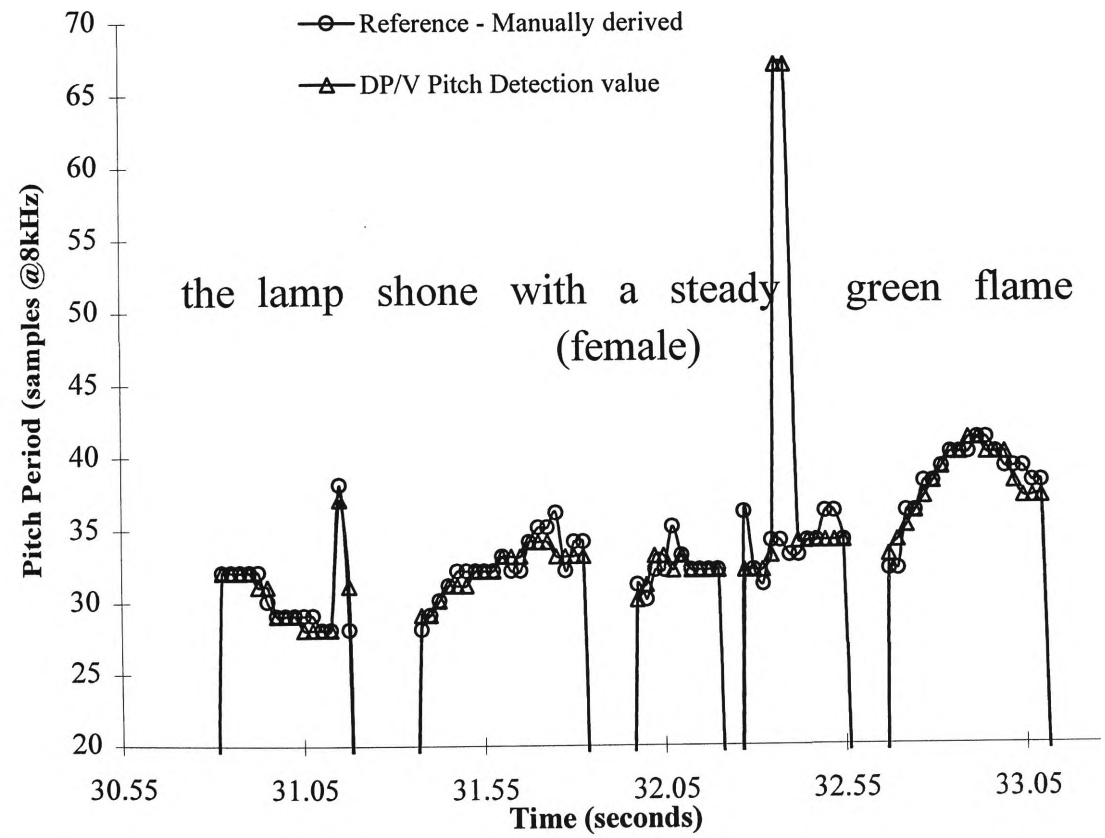

Figure D14 - DP/V Pitch Detector generated Pitch Profile

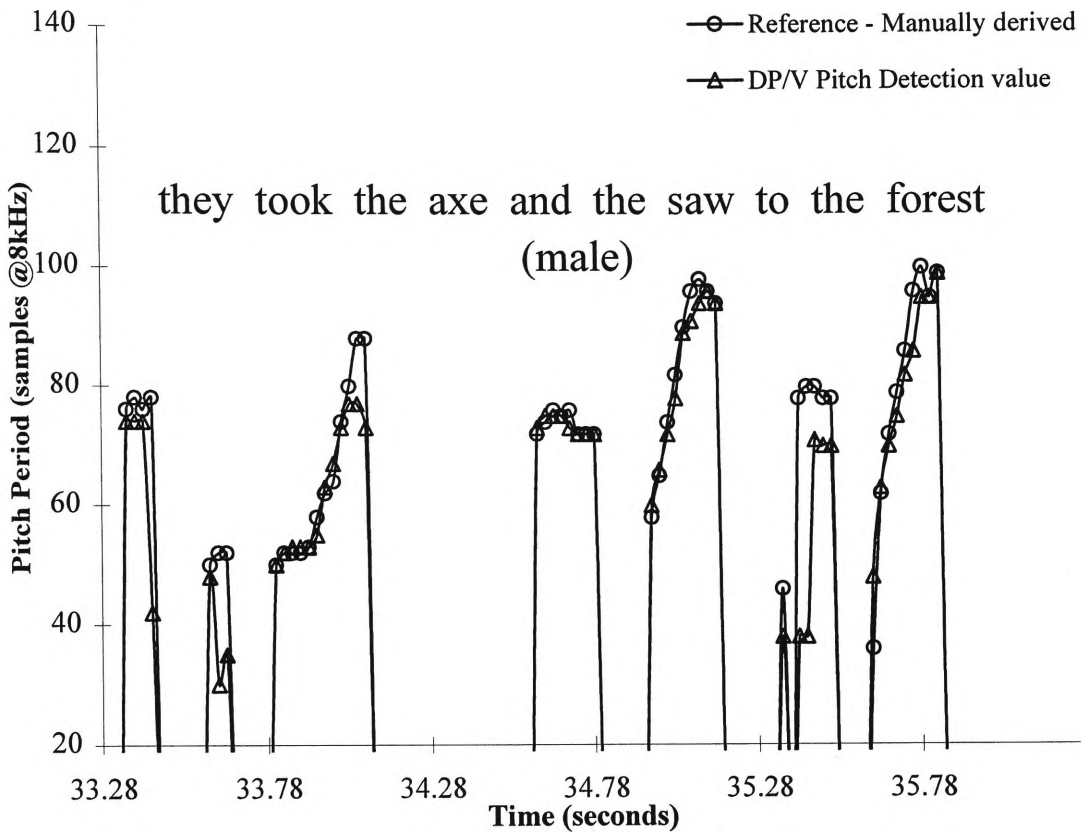

Figure D15 - DP/V Pitch Detector generated Pitch Profile

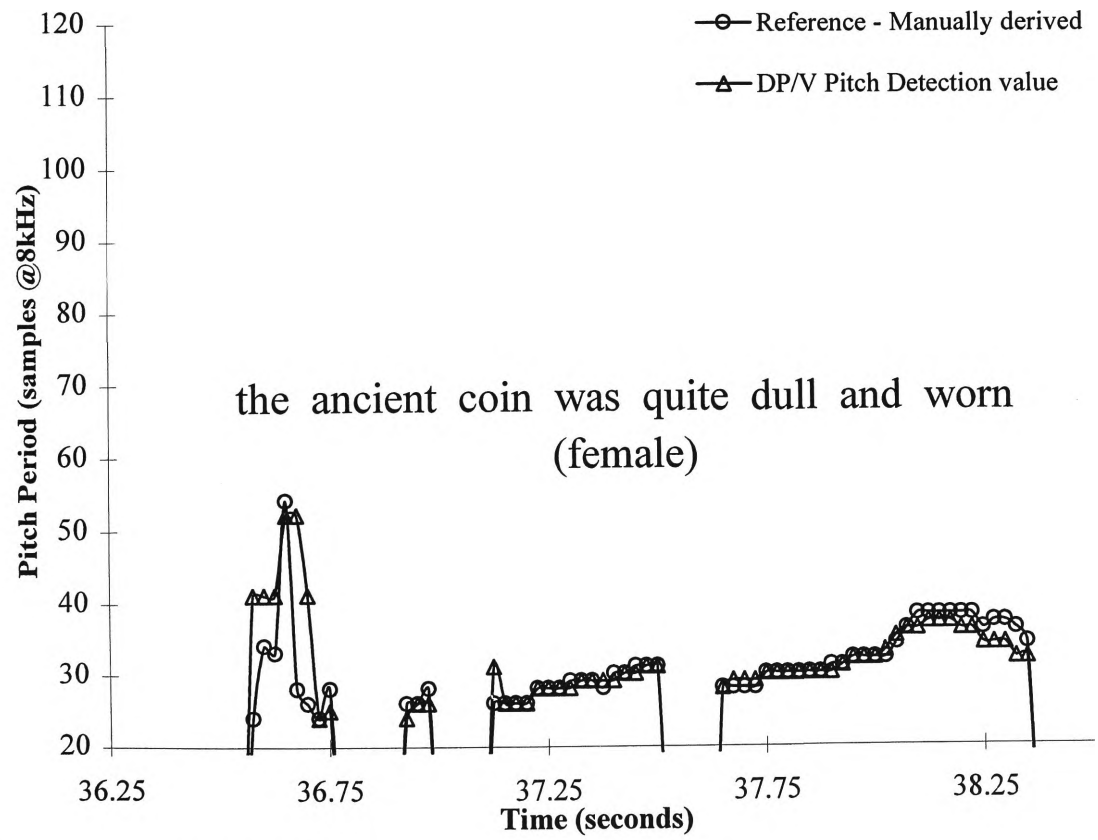

Figure D16 - DP/V Pitch Detector generated Pitch Profile

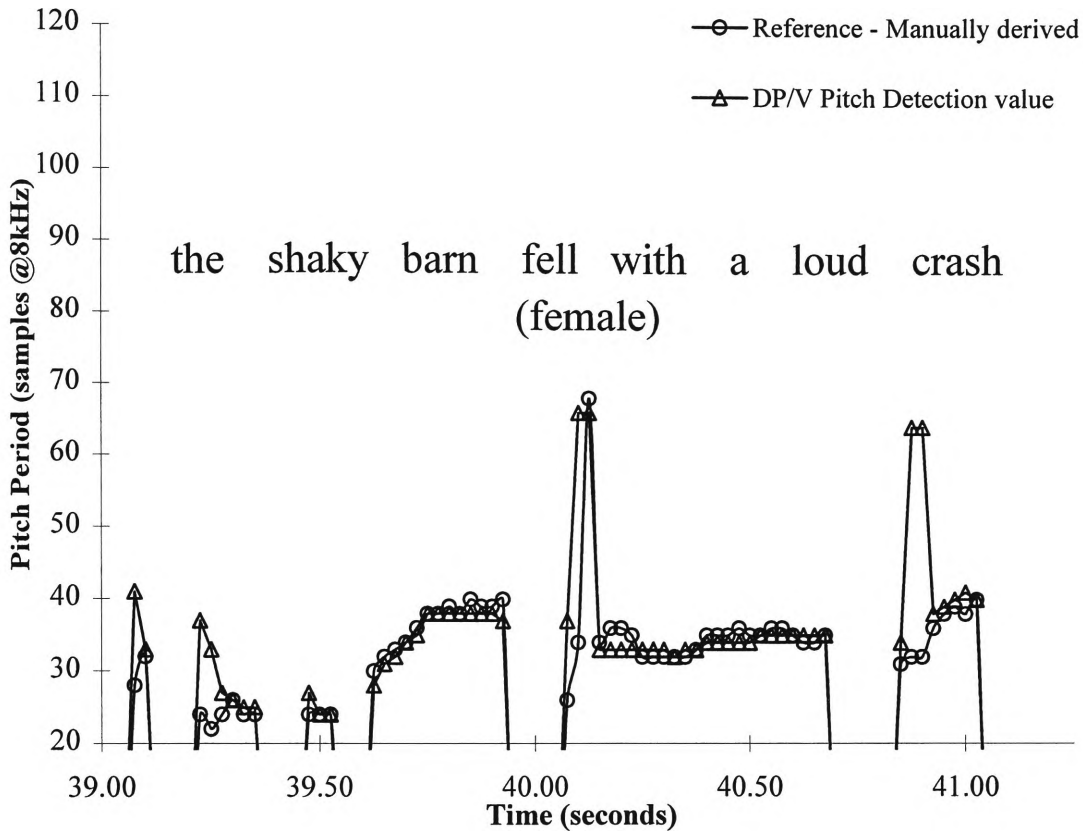

Figure D17 - DP/V Pitch Detector generated Pitch Profile

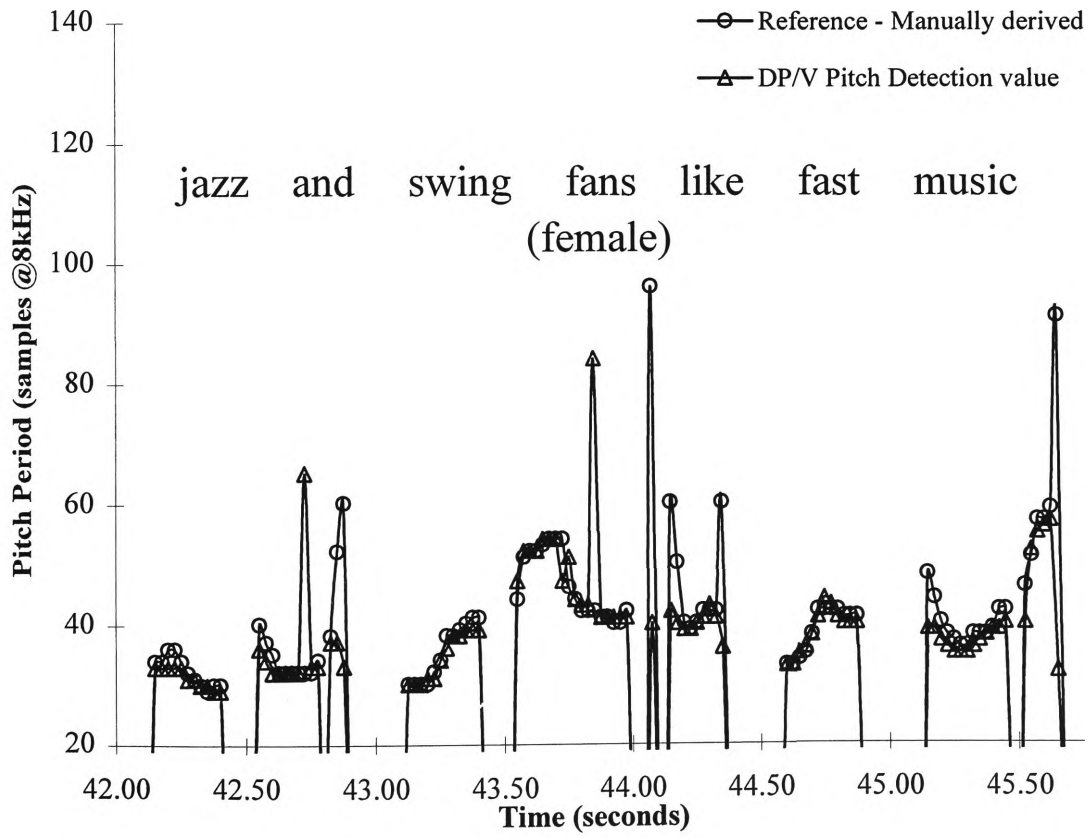

Figure D18 - DP/V Pitch Detector generated Pitch Profile

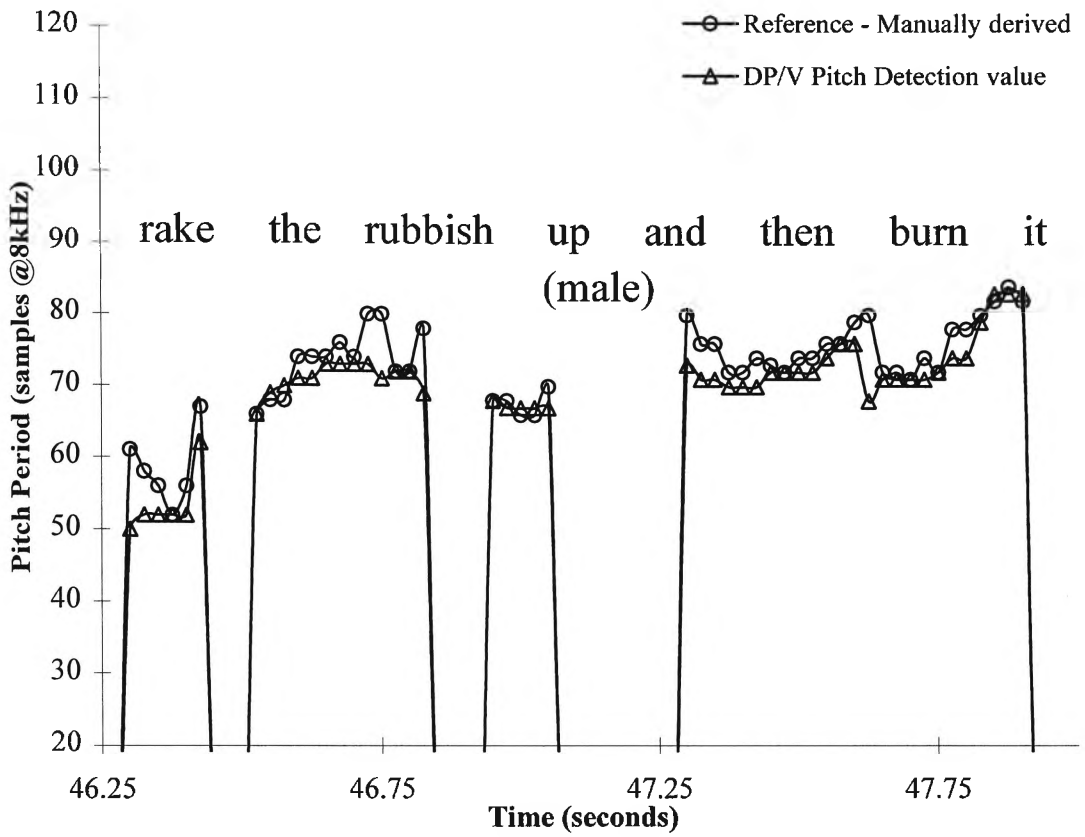

Figure D19 - DP/V Pitch Detector generated Pitch Profile

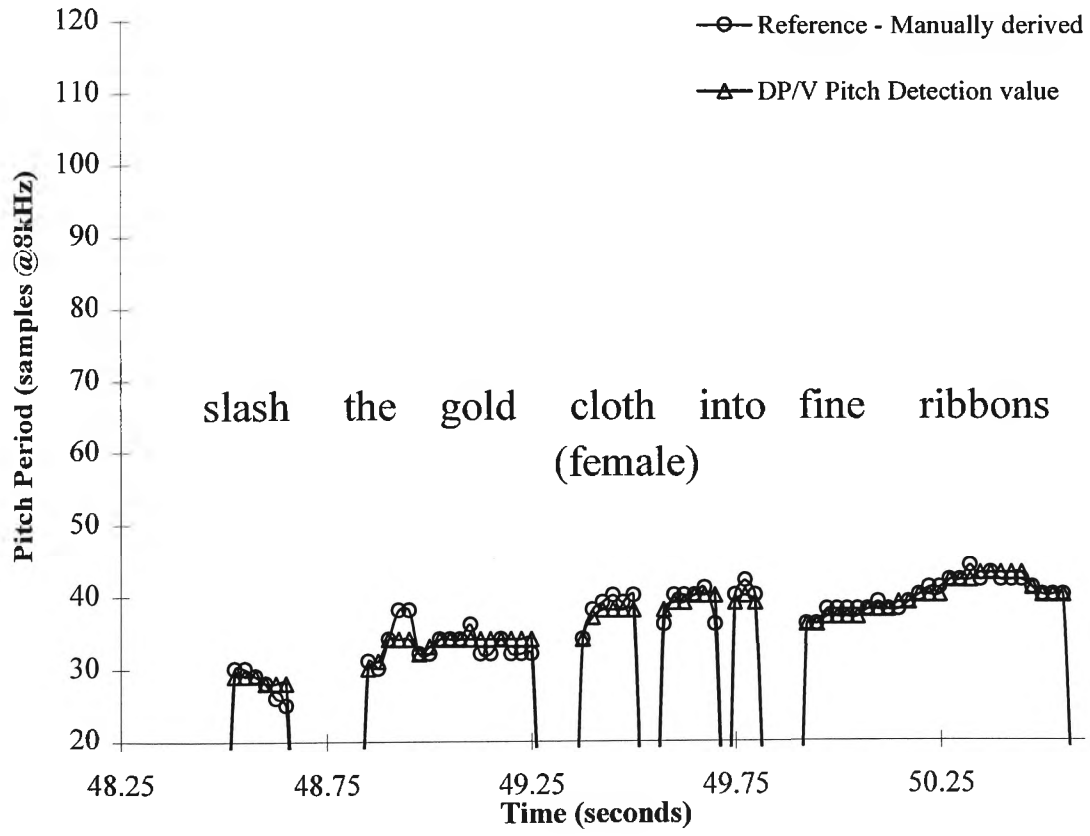

Figure D20 - DP/V Pitch Detector generated Pitch Profile